United States Patent [19]
Jones et al.

[11] 4,213,714
Jul. 22, 1980

[54] PRINTER HAVING VARIABLE CHARACTER DENSITY

[75] Inventors: Clifford M. Jones; William A. Surber, both of Waynesboro, Va.

[73] Assignee: General Electric Company, Waynesboro, Va.

[21] Appl. No.: 847,585

[22] Filed: Nov. 1, 1977

[51] Int. Cl.² B41J 3/12; B41J 19/32

[52] U.S. Cl. 400/124; 400/306; 400/322

[58] Field of Search ..... 364/900 MS File; 400/120, 121, 124, 303-306, 320, 320.1, 322, 328

[56] References Cited

U.S. PATENT DOCUMENTS

3,858,703	1/1975	Duley	400/124
3,905,463	9/1975	Boyce et al.	400/124
3,950,685	4/1976	Kramer	400/328 X
3,970,183	7/1976	Robinson et al.	400/124
4,034,842	7/1977	Giacone	400/322 X
4,050,563	9/1977	Menhennett	400/124

FOREIGN PATENT DOCUMENTS

1205744	11/1965	Fed. Rep. of Germany	400/124
2533312	2/1976	Fed. Rep. of Germany	400/320.1
2515557	10/1976	Fed. Rep. of Germany	400/124
2306835	11/1976	France	400/124

Primary Examiner—Paul T. Sewell
Attorney, Agent, or Firm—Michael Masnik

[57] ABSTRACT

An improved matrix printer is described employing a high order, single track incremental position encoder to provide a plurality of electrical reference signals representative of predetermined print positions along a platen and for locating matrix columns at the predetermined print positions. An electronic circuit is provided for varying the predetermined print positions at which the matrix columns can be located.

12 Claims, 7 Drawing Figures

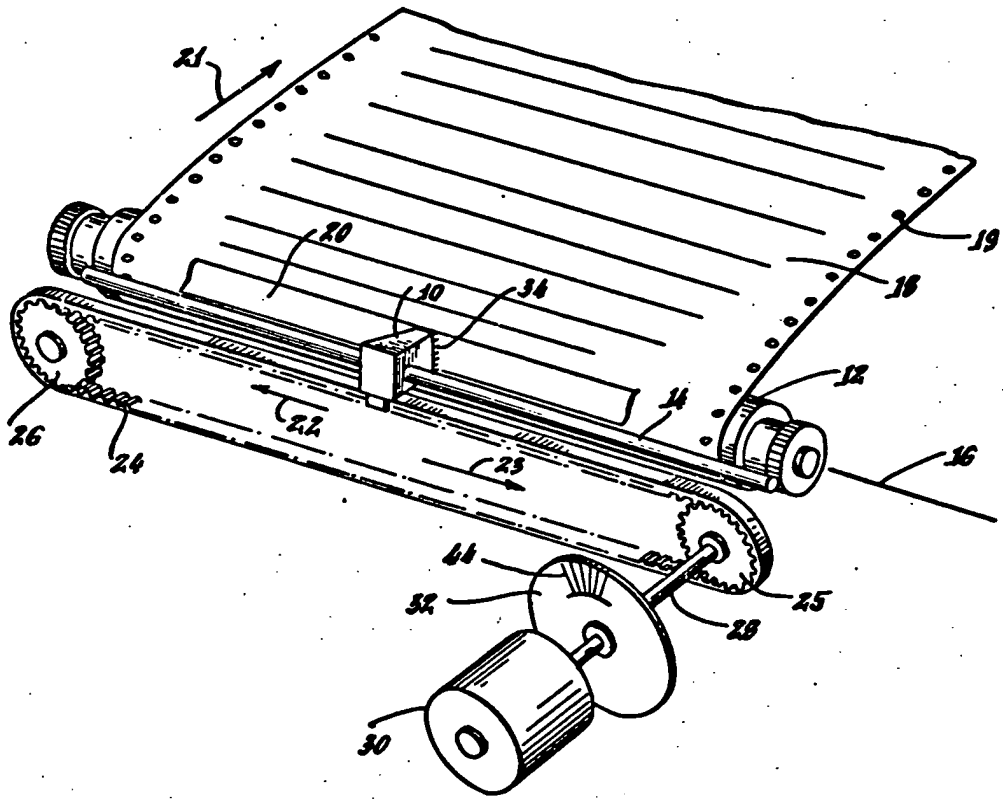


Fig. 1

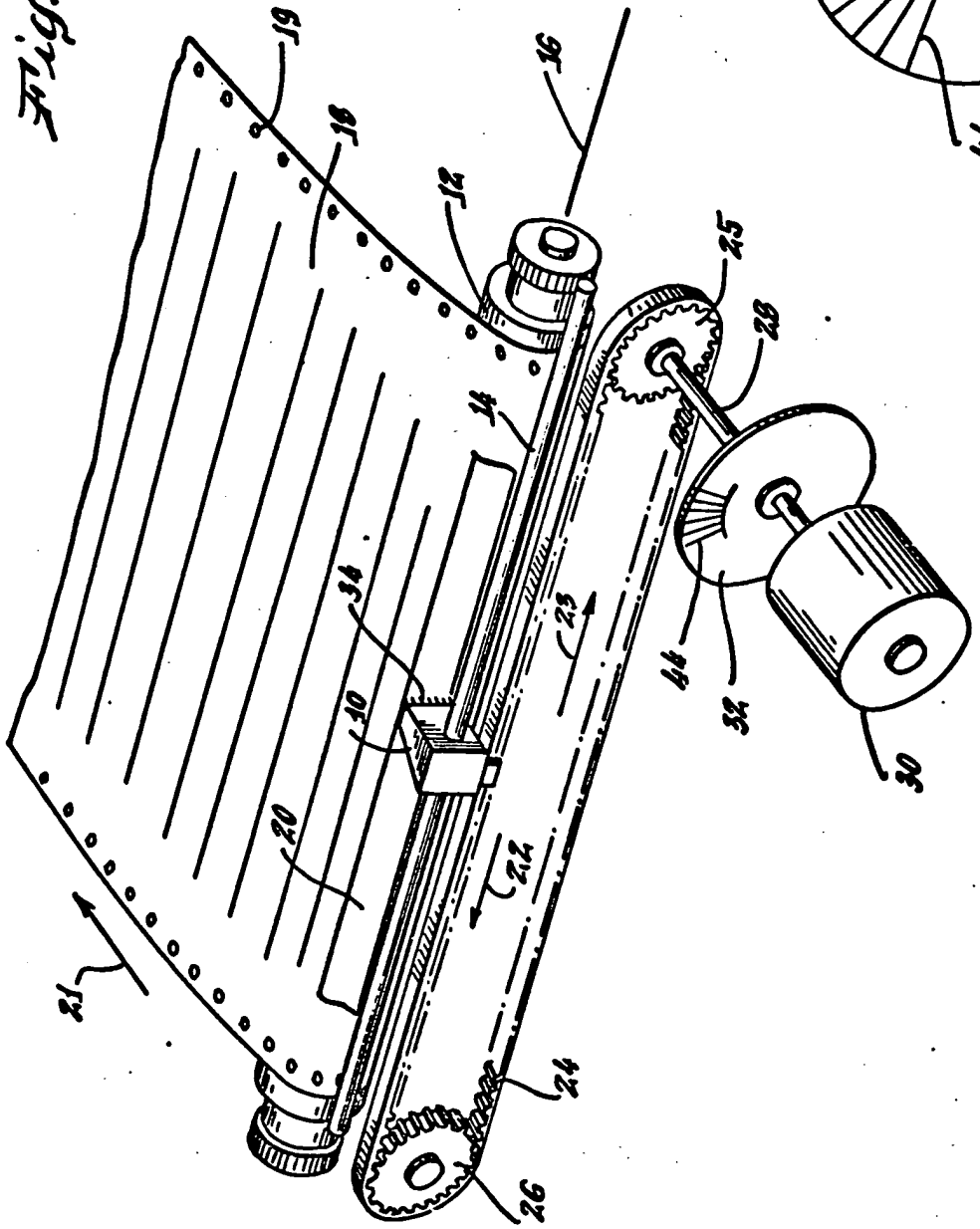
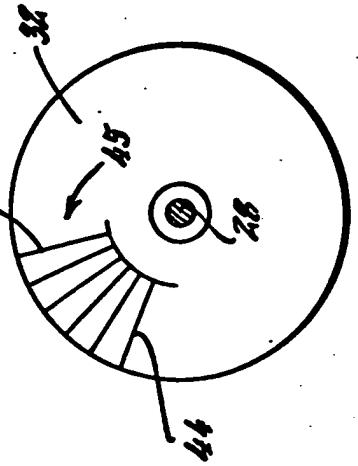


Fig. 2.



relatively smaller character print width and is less for a relatively greater character width.

In accordance with more particular features of the invention, a dot matrix, impact printer for reproducing characters of selectively differing character widths and wherein the character width of a character being formed is determined by spacing between adjacent matrix columns in a character matrix, includes an elongated platen and a printhead means which is adapted to print a matrix column of component character marks. Transport means advance the printer head means in a lateral direction, parallel to the platen. Means, including a body having a plurality of reference marks formed thereon, is provided for generating a plurality of electrical incremental position signals which are representative of print positions along the platen. A circuit means provides a character width signal which is representative of the character width of a character being printed. A circuit means which is responsive to the reference signals and to the character width signals selects print positions at which matrix columns are located for different character widths and provides electrical, matrix column signals timed in accordance with selected print positions. A circuit means is also provided which is responsive to the electrical matrix column signals and to a character signal for causing the printhead means to form a matrix column of component character marks at a matrix column location.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects and features of the invention will become apparent with reference to the following specification and to the drawings wherein:

FIG. 1 is a fragmentary, perspective view of a matrix impact printer with which the present invention is utilized;

FIG. 2 is a diagram illustrating an encoder disc used with the printer of FIG. 1;

FIG. 3 is a diagram illustrating a rectangular, dot matrix array for characters of a first character printing density;

FIG. 4 is a diagram illustrating a rectangular, dot matrix array for characters of a second character printing density;

FIG. 5 is a block diagram of an impact printer constructed in accordance with features of this invention;

FIG. 6 is a diagram of a logic array of FIG. 5; and,

FIG. 7 is a partial schematic, logic diagram for the array of FIG. 6.

DETAILED DESCRIPTION

Referring now to FIG. 1, an impact printer of the dot matrix type is shown to comprise a printhead means comprising a printhead 10 which is supported for transport adjacent a platen 12 on a track comprising a machined rod 14. The rod 14 is spaced in a transverse direction from the platen and extends in a general direction parallel to a longitudinal axis 16 of the platen. Positioned between the head 10 and the platen 12 are a record medium comprising an edge perforated, elongated, sheet 18 and an inked print ribbon 20. Conventional cog wheels, not shown, engage edge perforations 19 and provide for stepped advancement of the sheet 18 in the direction of the arrow 21. The printhead 10 is alternatively transported laterally along the track 14 in directions represented by arrows 22 and 23, parallel to the platen 12. It is advanced by a gear belt 24. The gear belt 24 is coupled to the head 10 and extends about a drive capstan 25 and an idler pulley 26. Rotating motion is imparted to the capstan 25 by a drive shaft 28 of a servo motor 30. An encoder disc 32, described more fully hereinafter, is mounted on the drive shaft 28 for rotation therewith. As the gear belt 24, which is mechanically coupled to the printhead and to the capstan 25 is rotated, the head 10 will be advanced along the rod 14 alternatively in directions indicated by the arrow 22 and 23, depending upon the direction of rotation of drive shaft 28. Advancement is continuous or alternatively is stepped. The head 10 will be actuated in a direction transverse to the platen axis 16 as it is being advanced in the direction of the arrow 23. This transverse actuation causes printing of component, character dots. Similarly, the head 10 will also be transversely actuated when it is advanced in the direction of the arrow 22 so that printing occurs when the head is transported in either direction.

The printhead 10 includes a vertically aligned array of print wires which is referred-to generally in FIG. 1 by the reference numeral 34. Each of the print wires of the array is selectively, electromagnetically energized by an associated solenoid winding, not illustrated. A print wire advances in a transverse direction with respect to the axis 16 of the platen and a leading segment of the print wire impacts the ribbon 20, the sheet 18 and the platen 12. Impact causes printing of an inked area corresponding to a face of the forward wire segment. One or more of the print wires are selectively energized to print character dots of a single matrix column. As the head is advanced laterally, the array of print wires are selectively energized to form one or more dots at successively located matrix column locations, thereby forming a dot matrix character.

Variable character width can best be appreciated by reference to FIGS. 3 and 4. These figures illustrate an area which is scanned by the head 10 during its lateral movement along the platen. Reference numeral 36 indicates a plurality of print positions and reference numeral 38 indicates matrix column locations. The reference numeral 38 indicates 9 matrix column locations in FIGS. 3 and 4. The area encompassed within the 9 matrix locations represents the width of the character matrix. It is apparent from these figures that character matrix is substantially wider in FIG. 3 than it is in FIG. 4 and that a same character printed in the matrix of FIG. 3 and of FIG. 4 will be substantially wider in FIG. 3 than in FIG. 4. The character formed in the matrix of FIG. 4 will provide greater character density since more of these relatively narrow characters can be fitted into a length of printing medium than can the character with the wider matrix of FIG. 3.

This compressability is accomplished in accordance with a feature of the invention by the generation of a plurality of incremental position signals which are representative of the predetermined possible print positions 36 along the platen. It is seen that the possible print positions 36 in FIGS. 3 and 4 are spaced the same distance apart and that the matrix column locations 38 of FIGS. 3 and 4 each located at one of the print positions. The character width is a predetermined integral multiple of the distance between adjacent print positions. For example, the character widths of FIGS. 3 and 4 are 48 and 32 print positions, respectively. The width may be larger, as indicated hereinafter, to provide for intercharacter spacing The distance D is, accordingly, a submultiple of the character width. The provision of a plurality of incremental position signals corresponding

to print positions along the platen enables repositioning of the matrix column locations at different print positions in order to compress or to expand the character size and therefore to reduce or increase the character density. Thus, in FIG. 3, the matrix columns are spaced by an equal multiple of print positions and a matrix column is located at every sixth print position. Similarly, in FIG. 4, the matrix columns are located at every fourth print position. Alternately, the matrix columns of a character are spaced by different numbers of print positions as for example, to enhance the appearance of the character. The incremental position signals provide for a first plurality of possible print positions intermediate adjacent matrix columns at a first character width and a second differing plurality of possible print positions intermediate adjacent matrix columns at a second differing character width. With reference to FIG. 3, five print positions occur between adjacent matrix columns for a character at a first character width. In FIG. 4, a second differing plurality of print positions occurs between the adjacent matrix columns. This second plurality comprises three print positions. Thus, by providing a plurality of electrical incremental position signals which are representative of relatively small, predetermined possible print positions along the platen, and by providing means for varying the print position at which a matrix column is located, variable width characters can be printed.

Referring now to the FIGS. 2 and 5, a means for generating the plurality of incremental reference signals which are representative of predetermined possible print positions along the platen includes a body comprising the disc 32, having a plurality of indicia 44 formed thereon, means comprising a head position transducer 40 for detecting passage of the indicia to provide incremental position signals, and phase to pulse converter circuit means 42 for providing incremental position signal outputs which are indicative of the rate and the direction of rotation of the disc 32. The indicia 44 of disc 32 comprise radially extending slits 44 which are arrayed in a single, circular track 45 about the disc. The head transducer 40 which is conventional and not shown in detail includes a source of radiant energy and is positioned for projecting radiant energy toward the slits of the rotating disc. The head transducer 40 also includes first and second photo detector means for sensing projected radiant energy and for generating first and second output signals. The photo detectors are spaced apart in a circumferential direction by an integral number of slits 44 plus ½ slit so that the detector signals are quadrature related and are utilized for indicating both rotation and the direction of rotation of the disc 32. These signals are applied over a line 46 to the phase to pulse conversion circuit means 42. The output of this converter, on line 48, comprises a first reference signal representative of rotation of the disc in a first direction and alternatively, on a second line 50, a reference signal representative of the rotation of the disc in a second oppostie direction.

The reference indicia 44 are formed to a relatively high resolution on the disc 32. Each slit then transmits radiant energy which is detected and generates a reference signal pulse which is representative of predetermined possible print positions along the platen. The relatively high resolution of the slits results in closely spaced print positions on the platen. In one example which is not deemed limiting in any respect, one inch of linear travel of the head 10 results in the generation of 660 print position signals. At a character density of 10 characters per inch, there are 66 reference signals and 66 corresponding possible print positions at which the nine matrix columns can be located.

The pulses from the head position transducer are applied to incrementing and decrementing terminals 52 and 54, respectively of a bi-directional, binary, reference counter 56. The output of the binary counter 56 is applied to, and is decoded by, a programmable logic array 58, described hereinafter. A second binary input to the programmable logic array 58 is provided from a character density signal source 59. A reset output is derived from the programmable logic array 58 on line 60 and is applied to the counter 56 for resetting the counter. Counter 56 will be automatically reset when the counter attains a count equal to a predetermined modulus for a character density of characters being printed. Character modulus represents the predetermined number of possible print positions assigned for printing the matrix columns of a character matrix for a selected character width. For example, a character density of 10 characters per inch will have a character modulus of 66. A character density of 15 characters will, for example, have a character modulus of 44. In the example given hereinbefore, 66 incremental position signals are generated for each character matrix for a character density of ten character per inch. The programmable logic array generates a reset pulse when the reference counter attains the count of 66. Similarly, lesser counts representing a smaller modulus for a character matrix at greater character densities will be cleared by a pulse on line 60 from the programmable logic array 58.

A diagrammatic representation of the programmable, logical array 58 is illustrated in FIG. 6. The array which comprises a logic matrix, is illustrated in more detail in FIG. 7. It includes a plurality of vertically enabled lines 62–82 representative of the character width and of the incremental position count of reference counter 56. The horizontal lines numbered 1 through 10 represent print positions 1 through 10 of a character matrix. It will be appreciated that the diagrammatic representation of FIG. 6 is for the purpose of illustrating that combinations of signal events from the reference counter and a character density signal results in particular outputs. In interpreting this representation, the x's at the junction of vertical lines 62–82 and the line print pos. 1 - print pos. 10 indicate possible print positions for the selected character density. It will be seen that at the density of 10 characters per inch of line 62, a matrix column is located at print position 1, 6 and 11. For purposes of simplifying the drawings, the entire array print pos. lines 1–66 are illustrated : id those horizontal lines between print positions 10 through 44 and 44 through 66 are represented by dotted line segments. It is apparent, however, that at a character density of ten characters per inch, and with a 9 column matrix, the matrix columns, which occur at every successive five print positions, will also occur at six successive print positions of 16, 21, 26, 31, 36 and 41. Similarly, at a character density of 15 characters per inch, the matrix column will be located at print positions 1, 5, 9, etc. As indicated hereinbefore, character appearance considerations may render it desirable to vary the inter-column spacing within a character. The spacing between matrix columns within a character can be varied to predetermined amounts. However, a matrix column 38 will be located at a predetermined possible print position and inter matrix column spacing will be

an integral multiple of the spacing ΔD between possible print positions. The reference counter 56 is automatically reset when the modulus for a selection character width is reached. In the case of 10 characters per inch, the modulus is selected to be 66. Since 45 space locations have been utilized in forming a dot matrix character at 10 characters per inch, the remaining space locations are utilized for intercharacter spacing. When the modulus for the selected character width is attained, a reset signal is generated on the reset line thereby resetting the reference counter for initiating the repeat of a countup of the character density modulus.

In the diagrammatic representation of FIG. 6, output column addresses for the first three matrix columns of a character matrix are represented at the intersections of the print pos. lines and those vertical lines 84–90 bearing the crosses. These four lines indicate which of the 9 matrix columns, at any one time, is selected for printing. The matrix column address will occur sequentially, depending upon the direction of head movement.

FIG. 7 is a logic diagram for implementing the arrangement of FIG. 6. The diagram of FIG. 7 illustrates logic circuit means for those functions indicated in the diagram of FIG. 6. It is understood that the diagrams of FIGS. 6 and 7 can be expanded in accordance with the schemes illustrated therein. The outputs from the position reference counter 56 are applied to AND gates 91 and 92 to indicate reference signals which are representative of print positions 5 and 10 of a character matrix. A reference counter output on line 94 is representative of print position 1. These signals are applied to AND gates 96, 98 and 100 along with the character width signal for a selected character density of 10 C.P.I. The outputs from gates 96–100 are indicative of print positions at which matrix columns are located. The outputs of these gates are applied to OR gates 102 and 104 to provide matrix column address information to a character ROM, discussed hereinafter. Similarly, AND gates 106, 108 and 110 are provided and enabled by reference counter outputs for print positions 1, 4 and 8 and by character width signal for the character density 15 C.P.I. The output of AND gates 106, 108 and 110 are signals indicative that the matrix columns are located at print positions 1, 4 and 8. These signals are applied to the gates 102 and 104 and provide matrix column address information to a character ROM. The logical array of FIG. 7 can be expanded to include the remaining matrix column locations for the character densities indicated and can be expanded to include other character densities. AND gates 112 and 114 are provided for generating reference counter reset pulses on output lines 116 and 118, respectively. These AND gates are enabled by the character density signals 19 and 10 and by the matrix modulus (print position) chosen for the particular character width. Additional reference character AND gates can be provided for other character densities chosen. The logic array of FIG. 7 can be provided by discrete circuit means or by integrated circuit means. It can also be implemented with memory devices, as for example, an ROM of American Micro Systems Inc. identified as AMI-S 8564.

The column address generated by the program logical array 58 is applied to a character ROM 120 along with an input signal from a source of character information 122. Print wires of the printhead array 34 of the printhead 10 are each individually energized by an associated solenoid to cause the printing of dots in a matrix column. The solenoids at any one matrix column location are selectively energized by information derived from the read only memory 120. One such ROM comprises AMI-S 8564 referred-to hereinbefore. The ROM 120 stores a 9 column dot format for each of the characters which can be formed by the printer. The ROM is addressed by a character code such as the standard ASCII code which is derived from the source of character information 122 comprising a communication data line or a keyboard switch for the printer. The ASCII code selects the particular character which is to be printed while the column address information from the programmable logic array 58 scans the 9 column format and indicates a particular matrix column to be printed at any one time. The output of the character ROM provides signals to print wire drivers 124 for energizing the solenoids 126 of a matrix column.

There has thus been described an improved printer having means for selectively and electrically varying the character width of characters being printed and density of characters on a line. The arrangement is advantageous in that it permits these character variations from an operator selectable keyboard source or alternatively from a communication line. Since the character variation is electrically selectable, the character width can be varied along any line being printed.

While there has thus been described a particular embodiment of the invention, it will be apparent to those skilled in the art that variations may be made thereto without departing from the spirit of the invention and the scope of the appended claims.

What is claimed is:

1. In a matrix printer for reproducing characters of selectively different character widths, said printer comprising head means which is transported in a lateral direction parallel to a platen, said head means adapted to form a matrix column of component character marks for forming a character on a record medium, a desired width of a character being determined by the spacing between adjacent matrix columns in a character matrix, the improvement comprising:

first means for generating a single set of electrical, incremental position signals representative of predetermined possible print positions which are all spaced apart an equal distance along said platen and for locating matrix columns at selected ones of said predetermined possible print positions,

second means coupled to said first means for electronically changing the predetermined print positions at which matrix columns are located, and

means for causing the distance between such selected adjacent print positions to always be a predetermined integral submultiple of the character width.

2. In a matrix printer for reproducing characters of selectively different character widths, said printer comprising head means which is transported in a lateral direction parallel to a platen, said head means adapted to form a matrix column of component character marks for forming a character on a record medium, a desired width of a character being determined by the spacing between adjacent matrix columns in a character matrix, the improvement comprising:

first means for generating a single set of electrical, incremental position signals representative of predetermined possible print positions which are all spaced apart an equal distance along said platen and for locating matrix columns at selected ones of said predetermined possible print positions,

second means coupled to said f[illegible]means for electronically changing the selected predetermined print positions at which matrix columns are located, and

means for causing the distance between such selected adjacent print positions to always be a predetermined integral submultiple of the character width, said electrical signals providing for a plurality of possible print positions between adjacent matrix column locations.

3. An improved matrix printer for reproducing characters of selectively differing character widths, the character width of a character being determined by spacing between adjacent columns in a character matrix, comprising:

(a) an elongated platen having a longitudinal axis thereof;

(b) a printhead means adapted to form a matrix column of component character marks;

(c) transport means for advancing said printhead means in a lateral direction, parallel to said longitudinal axis;

(d) means for generating a plurality of electrical incremental position signals representative of equally spaced possible print positions along said platen, said electrical signals provide for a first plurality of possible print positions between adjacent matrix columns for a first character width, and a second differing plurality of possible print positions between adjacent matrix columns for a second different character width;

(e) means for providing character width signals representative of a plurality of different widths for characters to be printed;

(f) circuit means responsive to said electrical incremental position signals and to said character width signals for selecting print positions at which matrix columns are located for different character widths and for providing an electrical matrix column signal timed in accordance with such selected print positions;

(g) means providing a character signal; and,

(h) circuit means responsive to said matrix column signal and character signal for causing said printhead means to print component character marks defining a character.

4. The matrix printer of claim 3 wherein said transport means includes a drive motor having a drive shaft thereof, said means for generating a plurality of electrical signals includes a body coupled to said drive shaft for movement thereof corresponding to movement of said drive shaft, said body having a plurality of reference marks formed thereon, said reference marks positioned in side by side relationship on a single track on said body and spaced apart for timing said electrical incremental position signals.

5. The printer of claim 4 wherein said body comprises a circular disc and said reference marks formed thereon comprise a plurality of radially extending indicia formed in said disc, said [illegible]icia positioned in a single circular track on said disc, means for detecting the passage of said indicia to provide detection signals, and means for generating electrical incremental position signals in response to said detection signals.

6. The printer of claim 3 including a reference counter adapted to count to predetermined different counts for different character widths.

7. The printer of claim 6 wherein said circuit means responsive to electrical incremental position and character width signals comprises a matrix logic circuit array.

8. The impact printer of claim 7 including a reference counter adapted incrementally to count within a predetermined modulus corresponding to a desired character width and wherein said matrix logic circuit array is adapted for generating a signal for resetting said counter means when said counter has counted to said predetermined modulus count.

9. The printer of claim 3 wherein said circuit means responsive to said matrix column and character signals includes electrical memory means for storing matrix patterns for a plurality of differing characters to be printed and said matrix column signals address successive matrix column patterns of a character to be printed.

10. The printer of claim 9 including printhead circuit driver means and said memory means provides output signals for enabling one or more of said output driver means.

11. Printer control apparatus adapted for operation with a printer of the serial dot matrix character printing type in which a printing head carrying selectively actuable matrix printing elements moves along the line to be printed for producing characters of selectively different widths in response to input signals representative of a plurality of different character widths for characters to be printed comprising a source of output pulses indicative of such head movement, a print control counter responsive to said output pulses for producing signals for controlling the print line locations where the actuation of printing elements is to be effected, means for causing the distance between adjacent print positions to always be a predetermined integral submultiple of the character width comprising said output pulses dimensioned to provide an available number of uniformly spaced output pulses per character which is greater than the maximum number of matrix element positions available to make up a character, and means for changing the count modulus of said print control counter in response to said input signals representing a change in character width.

12. The printer of claim 2 wherein a first plurality of print positions occur between adjacent matrix columns at a first character width, and, a second differing plurality of print positions occur between adjacent matrix columns at a second differing character width.

* * * * *

Fig. 3.

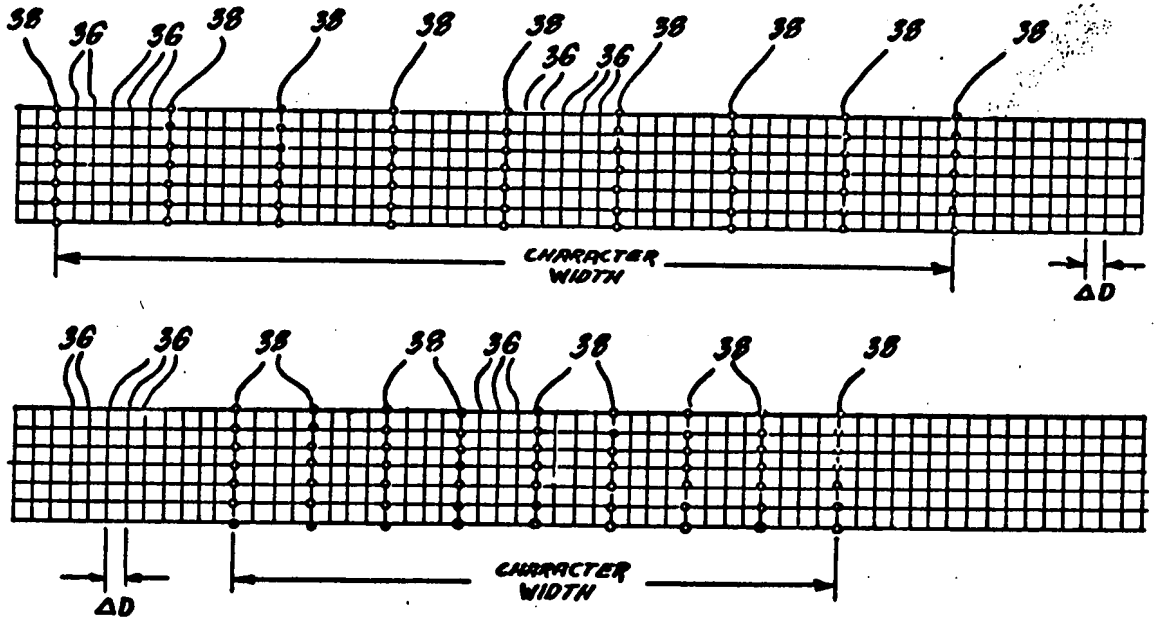


Fig. 4.

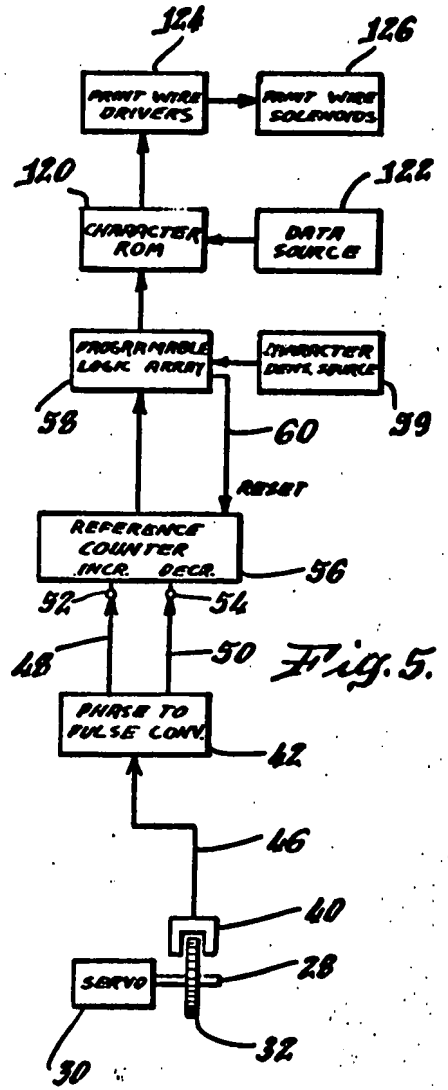


Fig. 5.

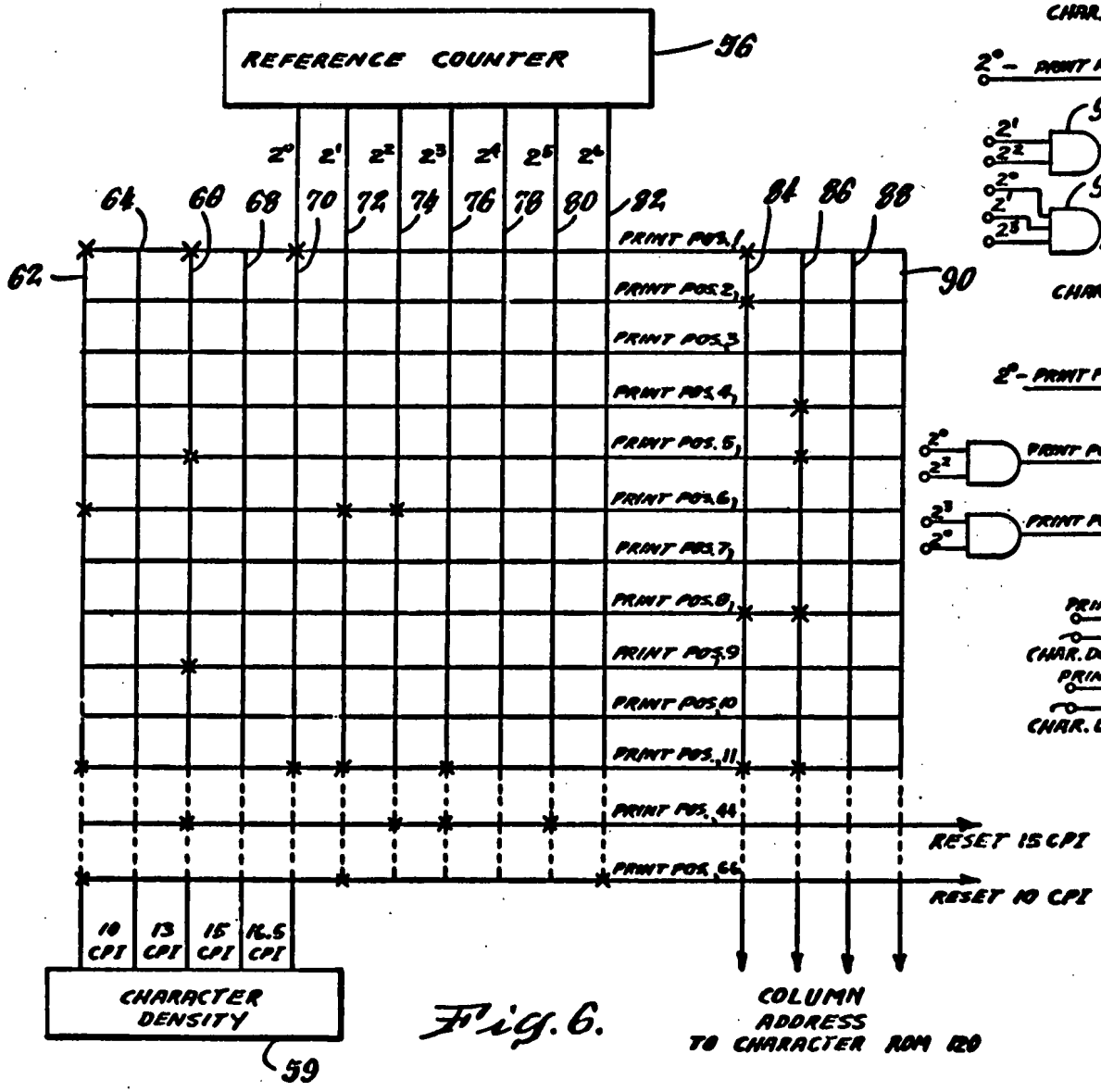

REFERENCE COUNTER
96
2^0 2^1 2^2 2^3 2^4 2^5 2^6
64 66 68 70 72 74 76 78 80 82 84 86 88
62
90
PRINT POS. 1
PRINT POS. 2
PRINT POS. 3
PRINT POS. 4
PRINT POS. 5
PRINT POS. 6
PRINT POS. 7
PRINT POS. 8
PRINT POS. 9
PRINT POS. 10
PRINT POS. 11
PRINT POS. 44
PRINT POS. 66
RESET 15 CPI
RESET 10 CPI
10 CPI
13 CPI
15 CPI
16.5 CPI
CHARACTER DENSITY
59
COLUMN ADDRESS TO CHARACTER ROM 120

Fig. 6.

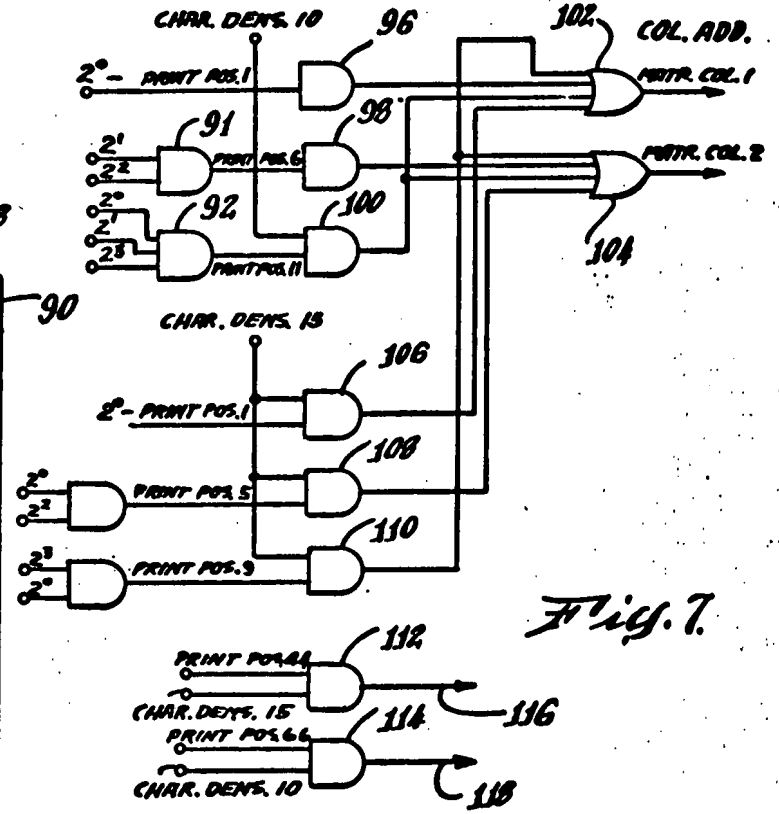

CHAR. DENS. 10
96
102
COL. ADD.
2^0 – PRINT POS. 1
ADDR. COL. 1
91
98
2^1
2^2
PRINT POS. 6
ADDR. COL. 2
2^0
2^1
2^3
92
100
PRINT POS. 11
104
CHAR. DENS. 13
106
2^0 – PRINT POS. 1
108
2^0
2^2
PRINT POS. 5
110
2^3
2^0
PRINT POS. 9
PRINT POS. 44
112
CHAR. DENS. 15
116
PRINT POS. 66
114
CHAR. DENS. 10
118

Fig. 7.

UNITED STATES PATENT OFFICE

CERTIFICATE OF CORRECTION

Patent No. 4,213,714 Dated July 22, 1980

Inventor(s) Clifford M. Jones & William A. Surber

It is certified that error appears in the above-identified patent and that said Letters Patent are hereby corrected as shown below:

Column 4, line 66, cancel "D" and insert -- Δ D --

Column 5, line 59, cancel "oppostie" and insert -- opposite --

Signed and Sealed this

Sixth Day of January 1981

[SEAL]

Attest:

SIDNEY A. DIAMOND

Attesting Officer

Commissioner of Patents and Trademarks

PRINTER HAVING VARIABLE CHARACTER DENSITY

BACKGROUND OF THE INVENTION

This invention relates to improved matrix printers. The invention relates more particularly to a printer which is adapted to print at variable character widths.

In one form of relatively high speed printing device, a character printing head is transported parallel to a stationary platen and is repeatedly accelerated in a transverse direction toward the platen. The printing head forms characters on the medium during the transport.

One form of relatively high speed printer comprises a dot matrix printer. In dot matrix printing, a character is formed by a plurality of printed dots which are selected from a rectangular array or matrix of dot locations arranged in columns and rows. The printing head in one form of dot matrix impact printer includes a plurality of individually selectable print wires which are aligned to form one or more of the dots of a matrix column. These print wires are accelerated toward the platen by associated solenoids at a matrix column location to cause printing on a record medium through an inked ribbon. Scanning of the head along the platen results in the successive columnar printing of additional dots necessary to form the characters at successive matrix locations.

The printing of a plurality of characters along a line requires that the position of the head be monitored and controlled. In dot matrix printing for example, the position of the array of print wires with respect to a dot matrix column should be established. To this end, head position encoders have been utilized which establish the desired positions along the platen of the dot matrix columns. This has been accomplished in one arrangement by utilizing an optical sensing system for measuring relative movement between the printhead and the stationary platen. In a particular optical sensing system, a circular disc having a plurality of incremental reference indicia, for example, lines or slits, formed thereon is mechanically coupled to a printhead drive motor. A stationary optical sensor is positioned adjacent the disc and radiant energy, which is projected from a source through incremental reference slits of the disc, causes the sensor to generate electrical signals representative of the position or printing positions and thus of the dot matrix columns to be printed. An alternative optical sensing arrangement utilizes a linear encoder instead of a rotating disc, linear fixed reference marks and an optical sensor which is mechanically coupled to and transported with the moving printhead. In either case, incremental reference signals are generated which are utilized for locating dot matrix columns of a dot matrix character.

The reference or position incremental signal has been utilized in several ways for locating the dot matrix column. In one arrangement, a reference incremental signal is generated which corresponds to each dot matrix column to be formed so that the signal itself is representative of the location of the dot matrix column. In another arrangement, a reference incremental signal is sensed at the location of an initial dot matrix column of a character. The printhead advances at a relatively constant printhead velocity and successive dot matrix column locations within the character are determined by timing the movement of the printhead with respect to the initial column. In still another arrangement, multiple encoder tracks are employed for separately sensing the initial and terminal locations of matrix columns of a dot matrix character.

The above-described matrix column location sensing arrangements provide for sensing the locations at a particular character density. For example, a presently utilized examplary character density is ten characters per inch. At times, it is desirable to increase the character density to relatively larger densities on a page. Character density can greatly be changed by varying the width of a character. Character width variation requires that spacing between adjacent dot matrix columns of a character be relatively changed. The reference incremental signals described hereinbefore are unsuitable to accommodate alternative printing at different character widths. In prior arrangements, the drive system was physically modified in order to provide for different character widths. For example, a drive wheel or gear was substituted in order to change the matrix column spacing for a character of different character width. Alternatively, an encoder disc was provided which included a plurality of reference tracks having reference lines and wherein each of the tracks was related to a particular character width.

The prior art arrangements for determining matrix column locations for different character widths exhibit several disadvantages. With gear or wheel changing, the operation of the printer is dedicated to a single character width. Relatively complex and costly circuit arrangements are required in the case of the encoder discs having multiple tracks.

Accordingly, it is an object of this invention to provide an improved matrix printer which is adapted to print at different character densities.

Another object of the invention is to provide an improved matrix printer having relatively non-complex and economical means for printing at a plurality of different character widths.

Another object of the invention is to provide an improved impact printer adapted to mix characters of different character widths.

A further object of the invention is to provide an improved means, including an encoder disc, for printing characters at different character widths.

SUMMARY OF THE INVENTION

In accordance with the general features of the present invention, a matrix printer is provided having means for generating a plurality of electrical incremental position signals representative of predetermined possible print positions along a platen and for locating matrix columns at the possible predetermined print positions. Means are provided for varying the predetermined print positions at which matrix columns are located. Means are also provided for causing the print positions to be predetermined integral multiples of desired character widths.

In accordance with other features of the invention, the incremental position signals provide plurality of print positions between adjacent matrix columns. A first plurality of print positions occur between adjacent matrix columns at a first character width, and, a second differing plurality of print positions occur between adjacent matrix columns at a second differing character width. The plurality of print positions occurring between adjacent matrix columns is greater in number for

United States Patent [19]

Buehner et al.

[11] **4,216,480**

[45] **Aug. 5, 1980**

[54] MULTIPLE SPEED INK JET PRINTER

[75] Inventors: **William L. Buehner; James D. Hill, both of Lexington, Ky.**

[73] Assignee: **International Business Machines Corporation, Armonk, N.Y.**

[21] Appl. No.: **960,417**

[22] Filed: **Nov. 13, 1978**

[51] Int. Cl.² **G01D 15/18**

[52] U.S. Cl. **346/75**

[58] Field of Search **346/75**

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,604,846	9/1971	Behane et al.	346/75 UX
3,723,646	3/1973	Behane et al.	346/75 X
3,878,517	4/1975	Kasubuchi et al.	346/75
3,928,718	12/1975	Sagae et al.	346/75 X
3,975,740	8/1976	Distler et al.	346/75
3,977,007	8/1976	Berry et al.	346/75 X

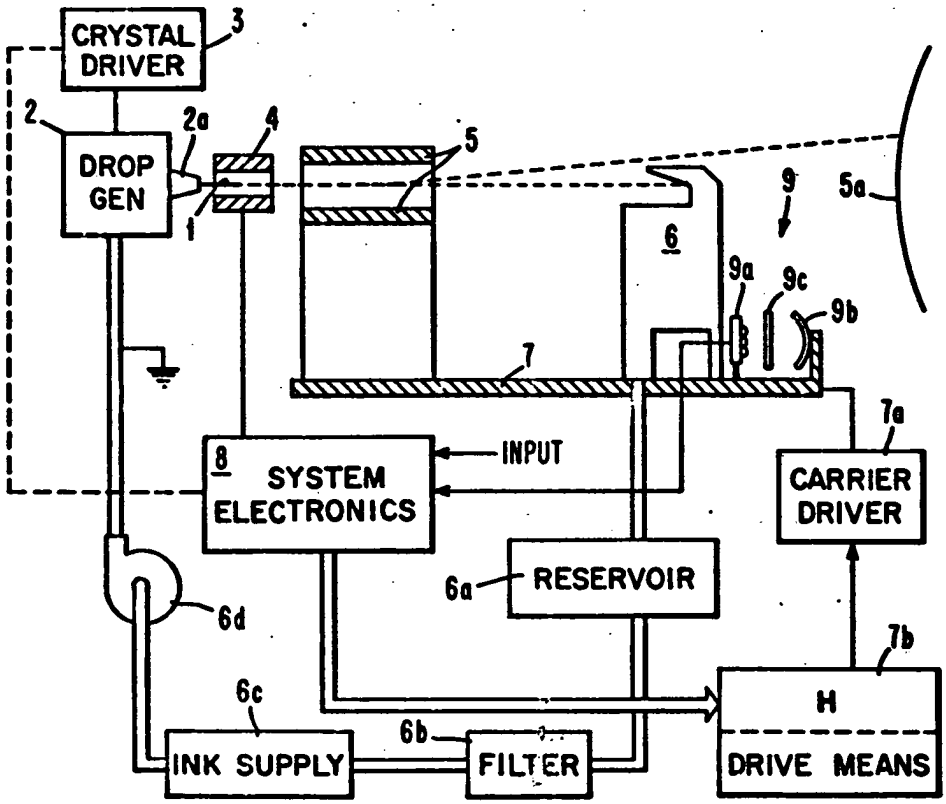
Primary Examiner—George H. Miller, Jr.

Attorney, Agent, or Firm—William J. Dick

[57] **ABSTRACT**

Disclosed is a multi speed ink jet printer in which alternate vertical scans of the ink stream may be omitted or alternate dots horizontally may be omitted or a combination of the two. In the alternate vertical scan omit mode, a factor of two increases in the speed of printing may be obtained, the same being true of a horizontal row dot omission. Moreover, a combination of the two permits up to a four times speed increase. In essence, while the rate of the ink stream remains constant, the predetermined maximum number of dots (picture elements) in a character box is decreased (resolution decreased) to permit an increase in the speed of printing.

20 Claims, 10 Drawing Figures



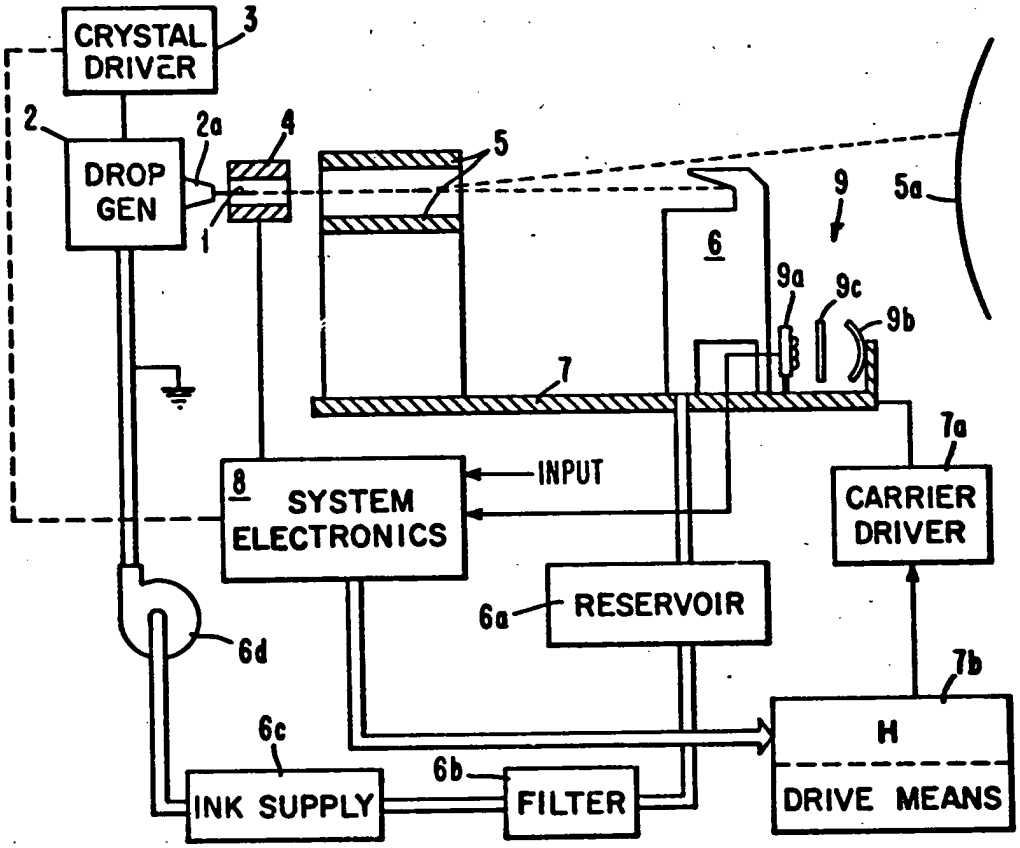


FIG. 1

NOTE: CIRCUIT SYMBOLS USED HEREIN ARE SHOWN BELOW

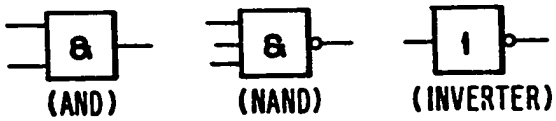


FIG. 1a

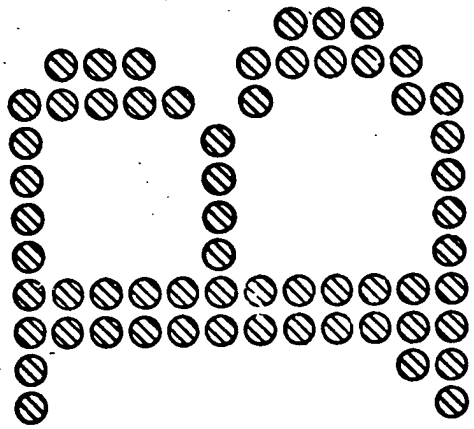


FIG. 4

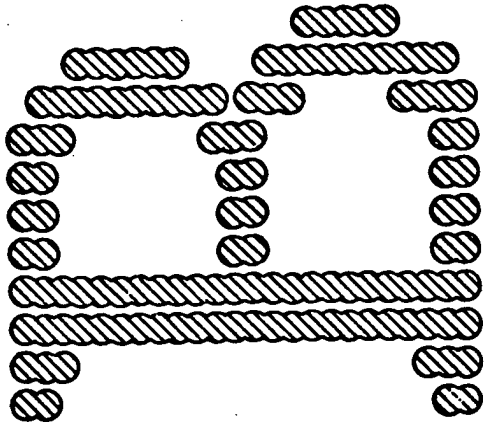


FIG. 3

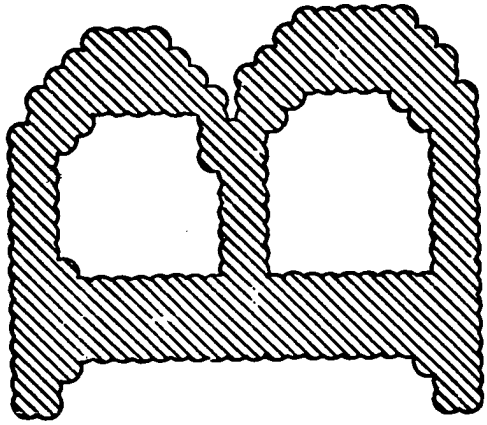


FIG. 2

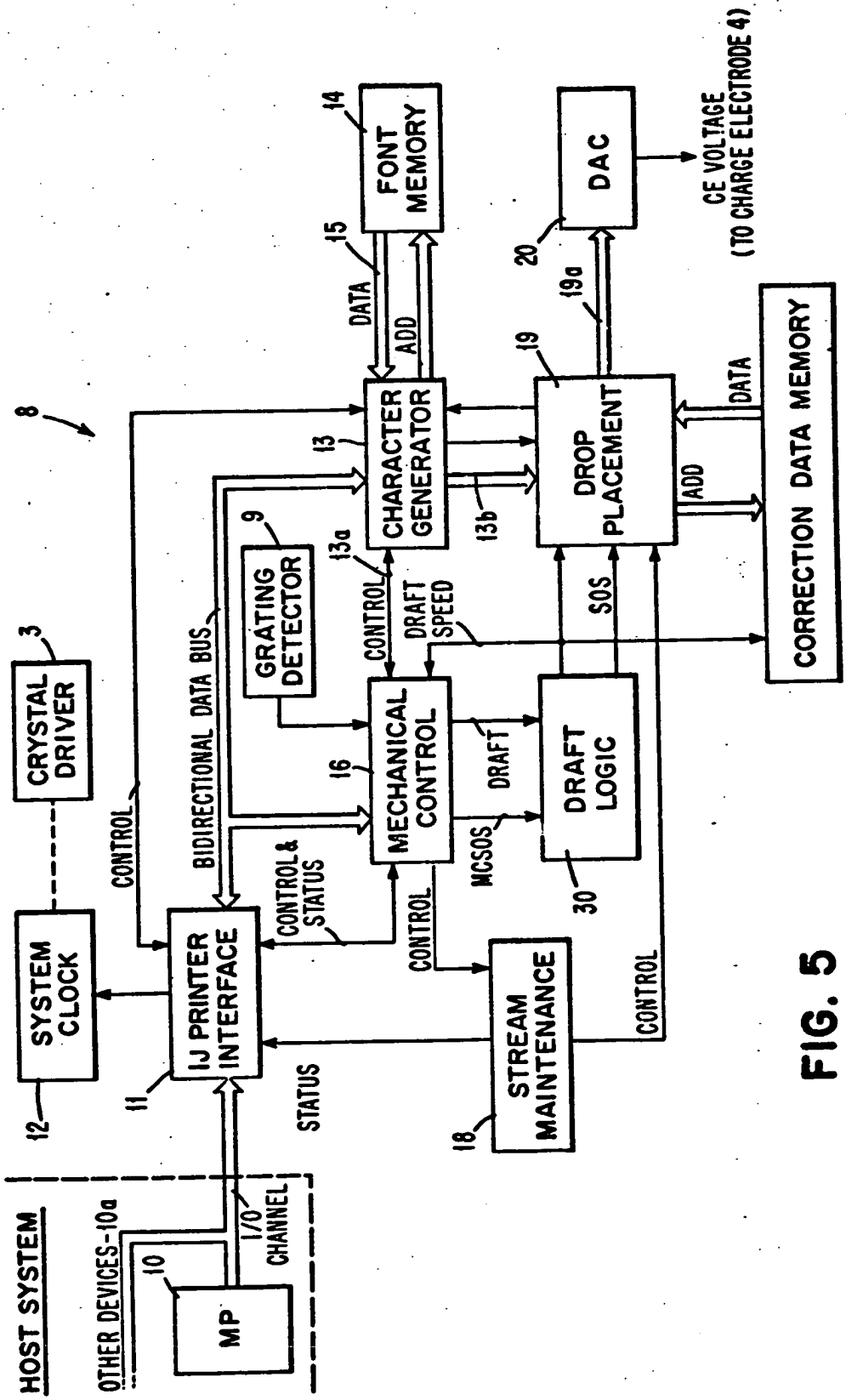

HOST SYSTEM
OTHER DEVICES-10a
MP
10
I/O CHANNEL
STATUS
11
IJ PRINTER INTERFACE
12
SYSTEM CLOCK
CRYSTAL DRIVER
3
8
CONTROL
BIDIRECTIONAL DATA BUS
CONTROL & STATUS
GRATING DETECTOR
9
13a
CONTROL
16
MECHANICAL CONTROL
MCSOS
DRAFT
DRAFT LOGIC
30
CONTROL
18
STREAM MAINTENANCE
CONTROL
13
CHARACTER GENERATOR
13b
DRAFT SPEED
SOS
DROP PLACEMENT
19
19a
20
DAC
CE VOLTAGE (TO CHARGE ELECTRODE 4)
DATA
ADD
CORRECTION DATA MEMORY
DATA
ADD
15
FONT MEMORY
14

FIG. 5

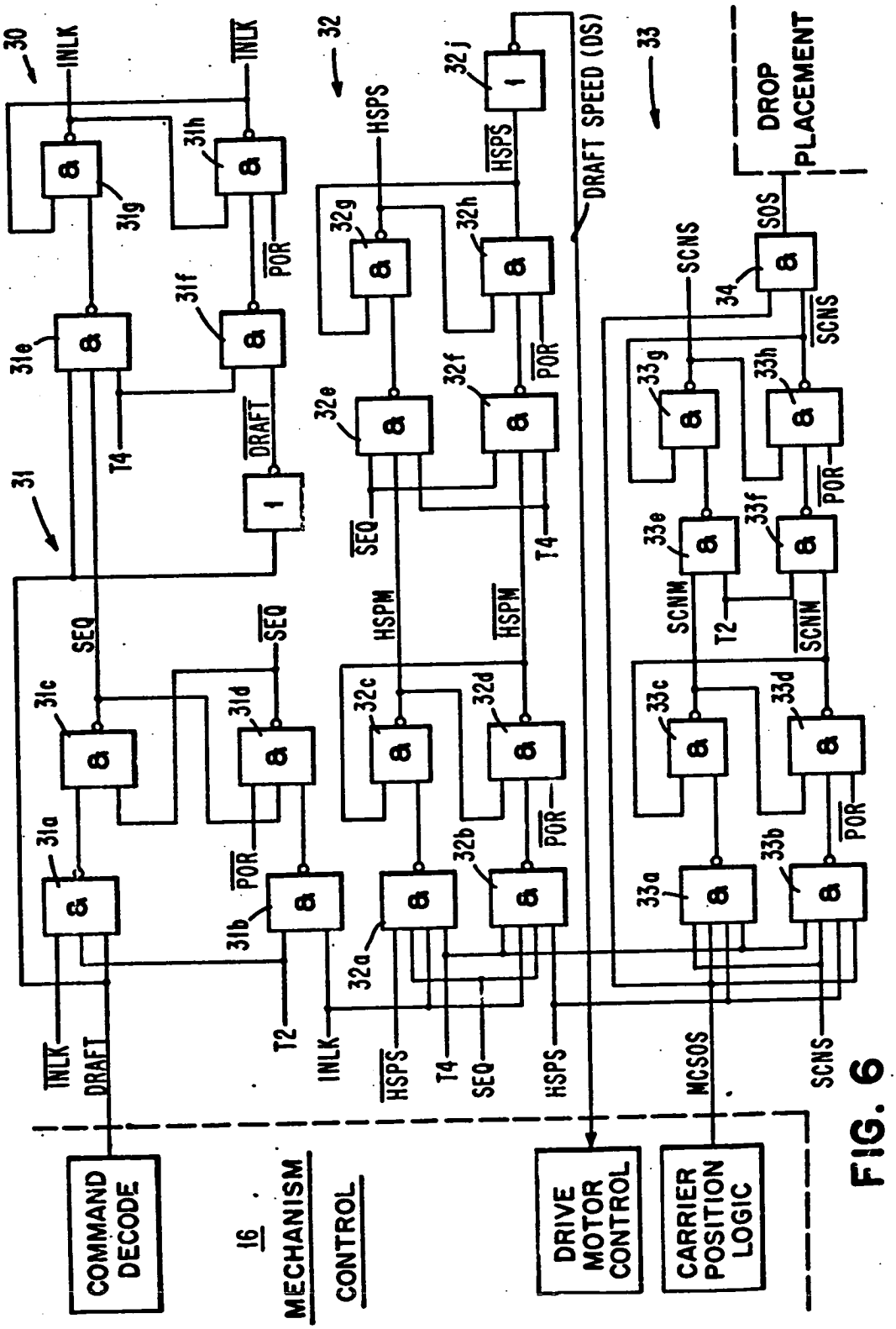

30
INLK
INLK
31h
31g
31f
POR
31e
T4
DRAFT
31
SEQ
SEQ
31c
31d
31a
POR
31b
T2
INLK
DRAFT
32
HSPS
32j
HSPS
DRAFT SPEED (DS)
32g
32h
32e
32f
POR
SEQ
T4
HSPM
HSPM
32c
32d
32b
POR
32a
HSPS
T4
SEQ
HSPS
33
DROP PLACEMENT
SCNS
SOS
34
SCNS
33g
33h
POR
33e
33f
SCNM
T2
SCNM
33c
33d
33a
33b
POR
MCSOS
SCNS
COMMAND DECODE
16
MECHANISM CONTROL
DRIVE MOTOR CONTROL
CARRIER POSITION LOGIC

FIG. 6

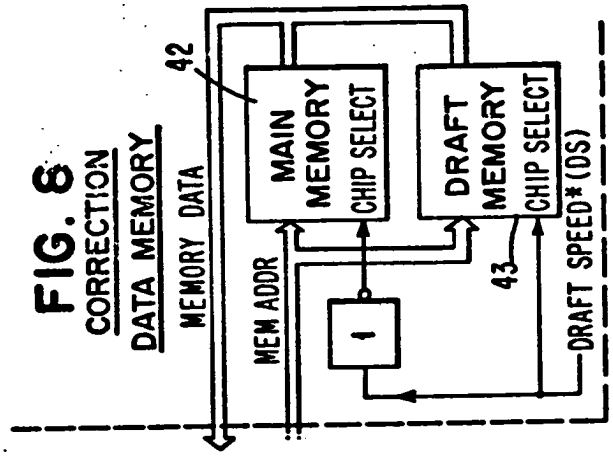


FIG. 6
CORRECTION DATA MEMORY

FIG. 7

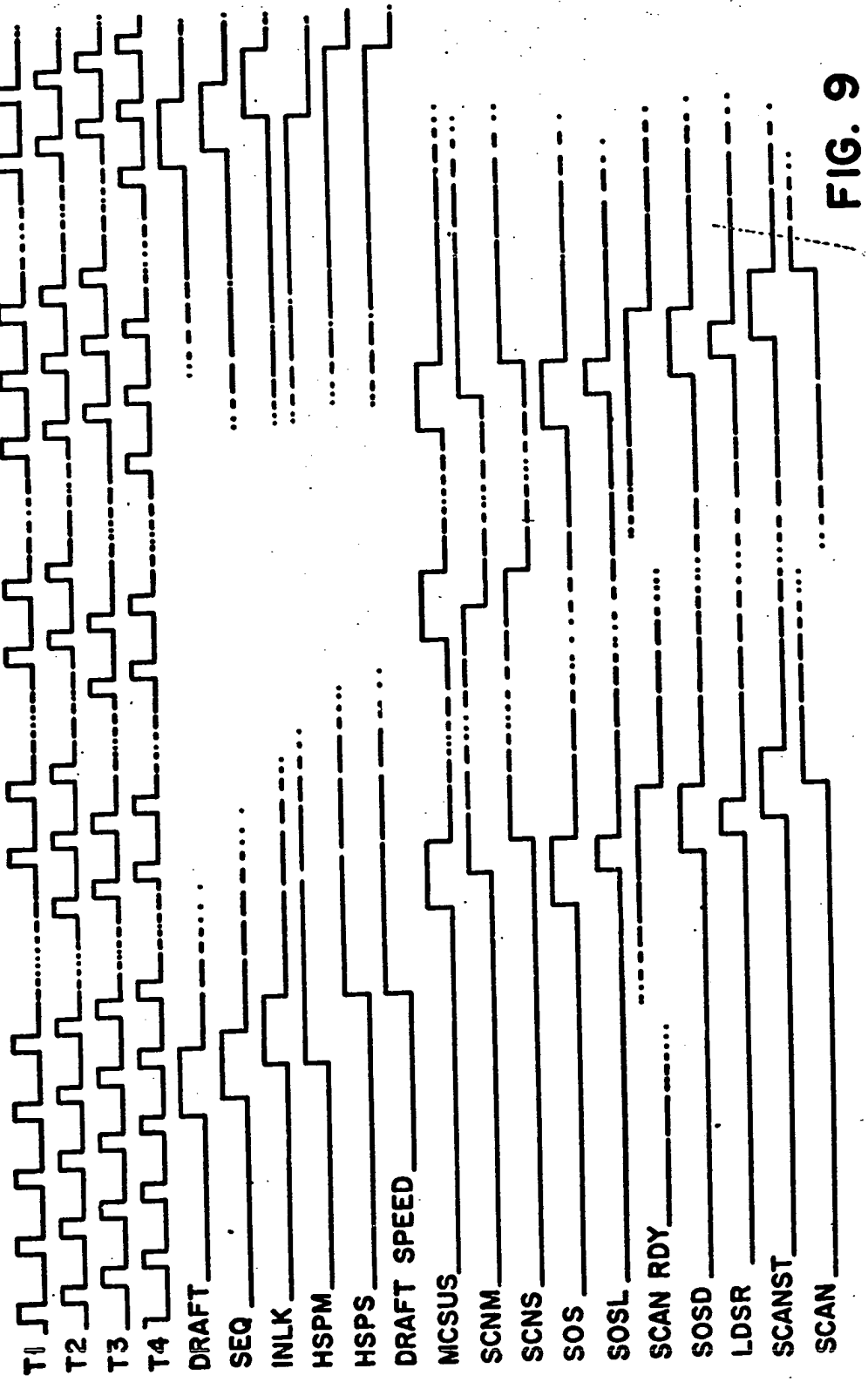

T1
T2
T3
T4
DRAFT
SEQ
INLK
HSPM
HSPS
DRAFT SPEED
MCSUS
SCNM
SCNS
SOS
SOSL
SCAN RDY
SOSD
LDSR
SCANST
SCAN

FIG. 9

MULTIPLE SPEED INK JET PRINTER

SUMMARY OF THE INVENTION

The present invention relates to ink jet printers of the Sweet type, and more particularly relates to an ink jet printer having provisions for operation at multiple speeds with a con-commitent change in character resolution.

STATE OF THE PRIOR ART

The IBM 6640 Ink Jet Printer which is utilized in IBM's 6640, 6650 Office System 6 is a continuous stream type ink jet printer based upon the Sweet principle as described in U.S. Pat. No. 3,596,275 issued on July 27, 1971 to Richard G. Sweet. In this system, a perturbated stream of ink drops is emitted from a nozzle under pressure, the stream breaking up into small drops at a predetermined point or distance from the nozzle. As the droplets are formed, they receive a charge in a charge ring or electrode and then pass between a high voltage pair of deflection electrodes, the amount of charge determining the deflection height of the ink droplets so as to form characters or other indicia on paper or the like adjacent to the discharge side of the deflection electrodes. If no indicia are to be formed, the drops receive no or little charge and as a consequence are not deflected when passing through the deflection electrodes, the drops then passing into a gutter for recirculation in the ink system.

As may be seen from the above, the vertical deflection of the drops is used to form a column of ink dots (called a scan) at the paper plane. Each successive drop in a scan is given a slightly greater charge so that the scan is constructed, in the example given, from bottom to top. Thus the characters are formed within a particular matrix by making a succession of scans. Typically the printhead elements are located on a movable carrier that travels in a predetermined path along the print surface cooperating with the vertical scan to provide the means for printing characters or other indicia on the print surface.

The IBM 6640 has a character resolution which gives a print quality substantially equal to typewriter print quality, the resolution being 94.5 pel/cm (240 pel/inch). A pel is defined as a picture element, in the instance of an ink jet printer a dot. For example, with a 94.5 pels/cm (240 pel per linear inch) resolution, and a character box 4.23 mm (1/6 inch) high (character matrix height) and employing a 10 pitch character, a dot matrix of $240/10=24$ pels wide (horizontal) and $240\times1/6=40$ pels or dots high may be formed. Thus a character matrix for each character in a 10 pitch character example, is 40 drops by 24 drops or dots or pels appearing on the paper. In this connection, in the IBM 6640, the relative dot placement tolerance is only plus or minus 0.033 mm (±0.0013 inches) while a character height and width tolerance of 0.076 mm (0.003 inches) is necessary to meet print quality requirements. Because of the gutter requirements, the character matrix height requires a maximum drop deflection of 0.51 cm (0.2 inches) since approximately 0.09 cm is needed to clear the gutter.

The IBM 6640 prints at a rate of approximately 90 characters per second, the resolution of the characters approaching that of high quality typewriter print. However, oft times it is not essential that the print be of such high resolution, especially if the document being generated by the printer is for "in house" use or, for example, for advertising circulars or mailers and the like.

In view of the above, it is a principal object of the present invention to provide an ink jet printer having the capability of varying its character resolution with a commensurate increase or decrease, as the case may be, in the speed of printing.

In the first embodiment, which may be termed the draft mode, there is a regular or periodic effective elimination of the start-of-scan signal, for example the nullifying of every other start-of-scan signal so that the printer only prints every other scan (column) of dots allowing the speed of the carrier to be approximately doubled. In this connection, alteration of the preset angle of the deflection electrodes may be avoided, if the tilt is objectionable when in the draft modes by employing the teachings of Ser. No. 864,066, filed Dec. 23, 1977 (now U.S. Pat. No. 4,138,688, issued Feb. 6, 1979) for automatically changing the tilt of the deflection electrodes dependency on carrier speed. In a second embodiment, every other dot in the scan is eliminated such that alternate rows of dots are missing. However, aerodynamic and drop interaction correction must be altered to compensate for skipping alternate dots or pels in a scan. Thus in the first embodiment, which may be termed scan skipping, while the width and height of the character box does not change, the maximum number of drops (pels) in each horizontal row with for example, a 10 pitch character is 12 (24/2), while the maximum number of dots vertically is still 40 for each scan. In the second embodiment, wherein every other horizontal row of dots is missing, the maximum number of dots in a character box is 24 (again utilizing the 10 pitch example) in the horizontal direction, but only 20 dots for each scan in the vertical direction, both embodiments permitting the speeding up of the carrier movement while maintaining the predetermined rate of the ink stream. Moreover, it is a simple matter to combine both scan skipping and horizontal row dot skipping so that the carrier may be speeded up to, for example, 4 times the quality mode speed. In this connection, minimal alteration of the existing hardware or electronics is necessary from the existing IBM 6640 for at least the scan skip draft mode.

In discussing drops and dots, the mean ink drop diameter in the IBM 6640 is 0.06 mm or 0.0025 inches.

In the quality mode of printing, the dots are printed on a matrix of 0.004 inches or 0.1 mm centers. Because of ink drop spread on the receiving medium (usually paper) the drops expand into dots approximately 0.006 inches or 0.15 mm in diameter, which means an overlap of the drops on the paper. In this context, in the first draft mode (preferred) of the present invention, the dots are on 0.008 inches or 0.2 mm centers horizontally and 0.004 inches or 0.1 mm centers vertically; in an alternative embodiment, i.e., horizontal row dot omission, the dots are on 0.004 inches or 0.1 mm centers horizontally and 0.008 inches or 0.2 mm centers vertically. Of course on the combination mode, the dots are on 0.008 inches or 0.2 mm centers both vertically and horizontally. In essence, the matrix resolution is sacrificed for a concomitant increase in speed, without changing or otherwise altering the predetermined rate of the ink stream.

With regard to the prior art, there are two references, notably U.S. Pat. No. 3,878,517 issued on Apr. 15, 1975 and U.S. Pat. No. 3,938,641 issued on Feb. 17, 1976 which present the closest prior art of which applicants

are at this time aware. The U.S. Pat. No. 3,878,517 relates to an ink jet system of the charge amplitude controlling type in which the printing velocity is controlled between a high velocity or low velocity mode, a control circuit within the character generator emits a signal which controls the drive rate to a servo motor connected to the carrier. Additionally, while the frequency of formation of ink drops remains constant, the control circuit, depending upon whether the apparatus is in a high velocity or low velocity mode emits a control signal to an adjustable frequency divider which changes the ratio of the number of drops receiving a video signal to the total number of drops formed to thereby compensate for changes in speed. In the printing apparatus disclosed herein, the ratio of charged to uncharged drops used in both what may be termed the draft and quality mode of printing remains essentially the same. There is no modification of the frequency of occurrence of the charging drops and the resulting ratio to the charged ink drops to the total ink drops in the stream by adjustment of any frequency division rate or of a frequency divider in response to a rate control signal.

The U.S. Pat. No. 3,938,641 describes a sequential dot matrix printer with a stepping type drive. The patent defines the controlling of the rate of column movement by changing the print rate signal. In either of the modes of operation heretofore described, there is no variation in the frequency to change the speed of the carrier or to change the speed of printing.

Other objects and a more complete understanding of the invention may be had by referring to the following specification and claims taken in conjunction with the accompanying drawings in which:

DESCRIPTION OF THE DRAWINGS

FIG. 1 is a fragmentary schematic view of an ink jet printer incorporating the subject matter of the present invention;

FIG. 1*a* is a symbology diagram indicating the symbols used in the circuits shown in FIGS. 6 and 7;

FIG. 2 is an enlarged character representation illustrating the formation of a character by the machine shown in FIG. 1 with a first resolution, or quality mode of operation, and with suitable grating signals below the character along with start-of-scan signal representations;

FIG. 3 is similar to FIG. 2 except showing the same character formed in a "draft" mode or second resolution;

FIG. 4 is similar to FIG. 3 except showing the same character formed in a "draft" mode as accomplished by an alternate embodiment of the present invention;

FIG. 5 is a schematic representation of the electronics of an ink jet printer incorporating the apparatus of the present invention;

FIG. 6 is a schematic logic diagram illustrating a first "draft" mode logic;

FIG. 7 illustrates the drop placement circuitry and logic diagram for the first "draft" mode logic;

FIG. 8 is a schematic diagram of correction data memory employed in the alternate "draft" mode;

FIG. 9 is a signal representation of a timing chart indicating the timing of the signals mnemonically represented in FIGS. 6 and 7 for proper operation of the printer in the first or preferred "draft" mode.

Referring now to the drawings, and especially FIG. 1 thereof, an ink jet printer is schematically illustrated therein. Basically, ink 1 under pressure is forced through a nozzle 2*a* as from a drop generator 2 to form a jet. While the jet would normally break up into a stream of drops of quazi random size and spacing, drop formation is controlled by vibrating the ink within the nozzle cavity at a fixed ultrasonic frequency as by a crystal driver 3, the crystal driver exciting a piezoelectric crystal within the drop generator 2. The pressure waves cause the jet 1 to break up into a stream of drops of uniform size and spacing at a well defined distance from the nozzle 2*a*. A typical drop generator structure is illustrated in IBM Technical Disclosure Bulletin, Vol. 21, No. 5, October 1978 at pages 1949, 1950.

A voltage applied to a charge electrode 4 surrounding the break up point (jet stream into drops) induces an electrical charge of a specific predetermined magnitude on the forming drop. This charge is retained by the drop throughout its flight to a drop receiving medium 5 such as paper and the like. The stream of drops passes through an electrostatic field formed by a fixed high voltage across a pair of horizontally disposed deflection plates 5. Because the charge on each drop is controlled individually, a drop can be deflected vertically a desired amount. In the instance of the IBM 6640 document printer, the drops are reflected vertically from bottom to top, one column of dots and/or spaces being referred to herein as a scan. If in forming a character, a particular space in a scan is to be left white, (unprinted) it is blanked by leaving the drops uncharged. These uncharged or undeflected drops are intercepted by a gutter 6 and recycled to an ink reservoir 6*a* through a filter 6*d*, an ink supply reservoir 6*c* and from there to the suction side of an ink pump 6*d*. The pump supplies ink on a continuing basis therefor to the drop generator 2. A typical ink recirculating system for ink jet printing apparatus is disclosed in U.S. Pat. No. 3,929,071, issued on Dec. 30, 1975 to the assignee of the present invention. Moreover, inasmuch as the height that the deflected drop will appear on the print receiving medium 5*a* is directly related to the residence time of the drop between the deflection plates 5 (that is for a given charge, the longer the resident time between the deflection plates 5 the higher the deflected drop) indicating that the velocity of the ink droplet or ink jet stream must be accurately controlled. In U.S. Pat. No. 3,787,882, issued on Jan. 22, 1974, is disclosed a number of servo systems for controlling the velocity of the stream by sensing the velocity and then servoing the pump to obtain the desired pressure within the drop generator 2. Another means of determining the velocity of the drop for obtaining servo control of the ink pump so as to control the velocity of the ink stream is disclosed in Ser. No. 843,081, filed on Oct. 17, 1977 to K. Meece et al.

The drop generator 2, charge electrode 4, deflection plates 5 and gutter 6 are all mounted on a carrier 7 which is driven horizontally along a predetermined print path, that is into and out of the plane of the paper in FIG. 1, at a relatively constant speed during the printing operation. In this manner, drops are deposited in appropriate positions within a character box or raster area to form the desired indicia or character. The carrier 7 is driven into and out of the paper by a carrier driver 7*a*, for example a DC motor in a controllable manner as by H drive means 7*b* (in the preferred embodiment) under control of the system electronics 8. The carrier driver 7*a*, H drive means 7*b* and portion of the system electronics 8 employed for movement of the carrier 7 into and out of the plane of the paper in FIG.

1 is fully set forth in co-pending patent application Ser. No. 954,374 of Morgan et al filed on Oct. 24, 1978, and incorporated herein by reference.

With the ink jet printer in operation, either in an interactive or on-the-fly or continuous printing mode, it is mandatory that some means be provided for indicating where the carrier is at any particular time so that the start of each scan may be determined in order that the charge electrode may receive the correct data or voltage level for proper deflection of the ink drops. Moreover, there must be some means provided for determining the direction of motion of the carrier. In this connection, a detector 9 which includes a light emitting and receiving matrix 9*a*, and a concave mirror 9*b* are disposed on the carrier on opposite sides of a fixed grating strip 9*c* (mounted on the machine frame), the grating strip in conjunction with the matrix and mirror permitting the output of signals to the system electronics for both charge electrode charging purposes as well as for controlling the H drive 7*b* and thus the carrier driver 7*a* connected to the carrier 7. In this connection, the detection apparatus and scheme is disclosed more fully in the Cialone and West patent application, Ser. No. 920,305 filed on June 28, 1978 and incorporated herein by reference. Moreover, the detection circuitry employed, while being disclosed in the aforementioned co-pending application, is also described in the co-pending application of Pettit, Ser. No. 920,306 also filed on June 28, 1978 and incorporated herein by reference. The grating 9*c* is a dual grating having opaque and transparent interdigitations, one grating portion being offset from another grating portion by 90°. The detector matrix includes a pair of light sources and a pair of detectors, light passing through the grating is reflected by the mirror 9*b* and impinges on the detectors which, with their associated circuitry output an alternating signal such as the output signals illustrated below the letter "B" illustrated in FIGS. 2, 3, and 4. (For purposes of identification, the two detector output has been labelled Det. A Det. B). One of the two detector outputs may be employed in conjunction with a counter to count the transistions (opaque to transparent), and by having a predetermined number of transistions on the grating thereby indicate absolutely the position of the carrier 7 at any point during its traverse along the print path. In the illustrated instance, in FIGS. 2, 3 and 4, the detector B output is utilized to indicate the position of the carrier and by simply interpolating the space of one cycle into four, the start-of-scan or SOS information may be derived.

The pulses indicated in FIGS. 2, 3 and 4 and labelled SOS indicate the start-of-scan of the ink stream. By way of example, if the grating associated with detector B output has 60 divisions per linear inch of grating (23.6 divisions per cm), and the resolution is 240 pels/inch (94.5 pels/cm) then each division of the grating must be further divided by four to meet the resolution requirements to form a character such as the letter "B" illustrated in FIG. 2. (The second detector output such as the detector A output is used for comparison with the detector B output to indicate to the electronics the direction of motion of the carrier).

In accordance with the invention, the resolution of a character may be changed to, for example, the resolution of the character illustrated in FIG. 3, the resolution being termed hereinafter draft mode 1. This is accomplished, in the instance of the resolution of the character "B" as shown in FIG. 3 by speeding up the carrier and simultaneously therewith eliminating every other start-of-scan (SOS) signal which is utilized to charge the charge electrode 4. A second "draft" mode alternative may be accomplished by skipping alternate rows of dots (horizontal rows) during the vertical scan. In this connection, correction of the normal aerodynamic and drop interaction repulsions must be taken into account, as by a modified correction memory. Alternatively, and as shown in FIG. 4, draft mode 1 may be further modified as illustrated in FIG. 3 by combining with "draft" mode 2 and skipping every other dot in a horizontal mode thereby resulting in the decrease in resolution illustrated in the character "B" illustrated in FIG. 4. In this mode, a speed increase may be realized of up to four times the speed of the printed character illustrated in FIG. 2 wherein a quality mode or quality resolution figure "B" is shown.

With the above background in mind, and turning first to FIG. 5, FIG. 5 illustrates the system electronics 8 including the apparatus constructed in accordance with the present invention. At the outset it should be recognized that the electronic diagram illustrated in FIG. 5 is substantially the same as the printer electronics utilized in the IBM 6640 ink jet printer with the exception of the function block labelled "draft logic" 30 and with some changes in the drop placement function block 19, both of which are discussed hereinafter relative to FIG. 6 (draft logic) and FIG. 7 drop placement logic.

Referring first to FIG. 5, a microprocessor or the like 10 with other input devices 10*a*, for example, tape, disc drive, typewriter, mag card etc. acts as a host system to the ink jet system illustrated in block diagram form in FIG. 5. The input to the printer is fed thereto by way of eight (in the illustrated instance) data lines, four control lines, an interrupt line and a master clock signal line making 14 lines which make up the I/O channel. The ink jet printer interface 11 provides, in a conventional manner, gating, logic, handshaking, suitable amplification, and an output from the master clock to the system clock 12 wherein frequency divider circuits divide the master clock frequency into the clock frequency T1–T4 illustrated in FIG. 9. The signals from the I/O channel may be suitably amplified and buffered so as to receive serial instructions from the microprocessor 10.

The data input into the printer system electronics for print commands is three 8 bit bytes, the makeup of which is more fully explained hereafter. By way of example, assuming that the writing line is a maximum of 14 inches long, and dividing each inch into 60 segments as has heretofore been described relative to the grating 9*c* (FIG. 1), is 840 segments (60 segments/inch × 14 inches) and a single byte of information will give only 2^8 or 256 bits which is insufficient. Therefore since at least 840 bits are required to give the position code to the printer in our example 14 inch line, and $2^{10}=1024$ bits is the closest to the actual position code, 2 bytes are necessary for command position signals from the processor to the printer. Moreover, in the actual system employed an additional 8 bits are required to identify the command signal as to the character to be printed, on the function to be performed and thus the printer interface 11 includes a buffer for those 3 bytes of information that come from the processor for driving the printer. In the 6640 ink jet printer, the first byte of a print command outputted on the IO channel by the microprocessor of the 3 byte signal is an address which is applied through the interface 11 to character generator 13 telling the character generator which character is to be

printed. (Assuming recognition as a print command). The character generator 13 outputs an address or a portion of the inputted address through the address bus ADD to the font memory 14 which permits data to be extracted from the font memory 14 through the data bus 15 and applied to the character generator. The print data extracted from the font memory 14 through bus 15 is data for a single vertical scan of the printer for the particular character which is to be printed.

The second and third bytes of the 3 byte signal from the microprocessor is applied to what, for lack of a better name, will be termed a mechanism control logic circuit 16 to give information to the mechanism control as to where the location of the first scan of the character to be printed is to be set.

The first byte of a function command is applied through interface 11 to the mechanism control logic circuit 16 to be decoded and used to initiate the appropriate action, for example the bit pattern 11010101 is decoded as a tab and 10110101 is decoded as a back space. The second and third bytes of the three byte function command from the microprocessor may also be applied to the mechanism logic control circuit 16 to be used to effect appropriate control actions when the carrier is at the location specified by these bytes, for example if a tab is decoded, the second and third bytes specify the stopping location of the carrier upon completion of the tab function.

Under the functional block mechanism control 16 may be included many functions. For example: (1) the control and timing of the stream maintenance 18 circuitry such as illustrated in the co-pending patent application of Neville and Taylor, Ser. No. 847,453, filed on Oct. 31, 1977, (now U.S. Pat. No. 4,136,345, issued on Jan. 23, 1979) and which is incorporated herein by reference, which monitors the ink stream at predetermined intervals to determine whether the deflected height of the drop is within tolerances; (2) to act as a command decoder for the sync and servo operations of the ink pump such as the servo control of ink pump illustrated in U.S. Pat. No. 3,787,882; (3) to control the carrier driver 7*a* and thus the horizontal velocity (in the present instance) of the carrier 7 as through H drive means, as with the aforementioned motor drive control set forth in co-pending application Ser. No. 954,374, filed Oct. 24, 1978 of Morgan, et al, and (4) to receive the grating position signals as from the detector 9 illustrated in FIG. 1 and discussed heretofore with particular reference to the grating detector and detection circuits of the co-pending Cialone, West and Pettit applications. Of course, the speed changer may take the form described in the IBM Technical Disclosure Bulletin, Vol. 20, No. 10, March 1978, pages 3993 and 3994. But the variable speed drive described in the Morgan et al application is preferred.

When the scan information derived from the font memory 14 has been loaded into the character generator shift register 39 (see FIG. 7), a signal goes out the control bus 13*a* to the mechanism control 16 which causes the mechanism control to put itself into a position ready to print, that is at the print position. With the scan information in the character generator 13, and a print signal being admitted to the mechanism control across control bus 13*a*, a scan ready signal is admitted from the character generator 13 through bus 13*b* to the drop placement logic circuitry 19. A second signal, from the mechanism control 16, start of scan (MCSOS) with coincidence of the carrier movement at a predetermined point on the grating 9*c* (FIG. 1), which information came from the second two bytes of the initial three byte signal applied by the microprocessor 10, effects an output of the drop placement generator 19 to a digital to analog converter 20 through bus 19*a* to apply charge electrode voltage to the charge electrode 4 (FIG. 1) of the printer, to charge the stream of ink drops to form the scan in accordance with the scan information emanating from the font memory 14 to the character generator 13.

The character generator 13 continues to load scan information from the character selected by the first byte of the three byte signal and continues to emit a print signal through control bus 13*a* to the mechanism control such that the cycle to print one character is continued until an end of character signal is emitted from the character generator through the bus 13*a* to the mechanism control circuitry 16. Additionally, while it requires coincidence of two signals to the drop placement logic circuitry or logic 19 from the character generator 13 and from the mechanism control circuitry, i.e., a scan ready and an MCSOS (which is converted as will be shown hereinafter to an SOS or start of scan signal), the drop placement circuitry provides a further signal which holds the signals until it is ready to accept the new data from the character generator which is utilized to charge the charging electrode of the printer. Moreover, if new information, for example another character is to be printed immediately after the first one, the print signal on control bus 13*a* is designed to override the end of character signal so as to continue to output the scan ready signal, withdraw the information from font memory, etc. necessary to provide the charging values to the charging electrode. In this connection, it should be noted that the font memory 14 may include a multiplicity of read only memory structures, which font to be selected being dependent upon the input from the microprocessor 10. A scheme for changing fonts by simply selecting different memory portions is illustrated in U.S. Pat. No. 3,964,591 issued on June 22 1976.

In the quality mode of printing to form characters such as the character "B" illustrated in FIG. 2, the machine operates essentially as set forth above, and this is the system, as described, for the IBM 6640 Document Printer.

Many times, however, it is unnecessary to provide a print resolution as fine as the high quality mode of printing illustrated in FIG. 2, being desirable to provide faster copy than is possible in the high quality print mode which is drop rate limited. Accordingly,, a draft mode of printing or mode 1 to produce characters such as illustrated in FIG. 3 such as the character "B" may be desired. To this end, and turning now to FIG. 5, a draft mode logic circuitry 30 is positioned intermediate the mechanism control circuitry 16 and the drop placement cicuitry 19 such that when printing is to take place as indicated by the suitable print function command from the microprocessor 10, in the quality mode, the circuitry 30 will permit direct access of the drop placement circuitry 19 by a start of scan signal (SOS) which is really MCSOS, bypassing the draft logic circuitry 30. Alternatively, when the microprocessor input to the printer system indicates that a draft mode quality is desired, the draft logic circuitry 30 is brought into play.

To this end, and referring now to FIG. 6, with a signal (a portion of the three byte signal coming from the microprocessor 10) decoded in the mechanism control indicating the speed of the machine is to be in-

creased and the resolution is to be decreased to the draft mode to form characters such as illustrated in FIG. 3, a command decode logic of for example 11110100 produces the first signal (DRAFT), the first occurrence indicating a shift to draft mode. The DRAFT signal (See FIG. 9) is applied to an input of a logic master slave flip flop or latch pair 31 to produce several signals at various clock times T1-T4 as illustrated in the designated clock sequences shown in FIG. 9. As illustrated in FIG. 6, the master slave flip flop or latch pair 31 is composed of NAND gates, in the particular embodiment shown, NAND gates 31*a*–31*h* and a single inverter 31*j* to produce the signals interlock (INLK), not interlock ($\overline{INLK}$), sequence (SEQ), not sequence ($\overline{SEQ}$), and not draft ($\overline{DRAFT}$). The power on reset (POR) or not power on reset ($\overline{POR}$) signals are the conventional signals derived from the logic utilized for initializing the machine at its start up. As shown, the signal inputs to the master slave flip flop 31 are the clock pulses T2 and T4 as well as the first or draft signal DRAFT from the command decode section of the mechanism control circuitry 16. As illustrated, these signals are coupled into a logic second master flip flop or latch pair 32 which produces as an output a draft speed signal DS which is applied as an output to the drive motor control such as illustrated in co-pending patent application Ser. No. 954,374 and incorporated herein by reference heretofore to speed up the carrier driver or drive motor 7*a* associated with the carrier 7 of the printer.

As illustrated, the latch pair or master slave flip flop 32 is also composed of NAND gates 32*a*–32*h* with an inverter 32*j* at the output of the master slave flip flop. Some of the inputs to the master slave flip flop 32 are generated by that latch pair, for example high speed pick master (HSPM) and not high speed pick master ($\overline{HSPM}$) which results in the high speed pick slave (HSPS) and not high speed pick slave ($\overline{HSPS}$) signals which are returned as inputs to the NAND gates 32*a* and 32*b* along with clock pulse T4. The SEQ,INLK signals are taken from the master slave flip flop or latch pair 31.

A third master flip flop or latch pair 33 receives a mechanical control start-of-scan signal (MCSOS) from the carrier position logic which is part of the mechanism control 16. As heretofore explained, the MCSOS signal is derived from the detector 9 (see FIG. 1) and its associated circuitry. Moreover, as will become more clear hereinafter, the MCSOS signal and start of scan signal (SOS) are one and the same when the printer is in the quality mode of printing, that is to form characters such as shown in FIG. 2. As described heretofore, the MCSOS signals generated are a multiple of the number of lines on the grating. Thus in the example given, if the grating lines are 60 lines per inch (23.6 lines per centimeter), 240 pulses (4 times 60) will be generated for each inch of carrier travel (94.5 MCSOS pulses every centimeter of travel). As the carrier speeds up, of course, the repetition rate of these pulses increases, but the distance traveled which is representative of each of the pulses will always remain the same. In this connection, the master slave flip flop or latch pair 33 acts as a divide by two circuit generating a scan master (SCNM) and scan slave (SCNS) signal as well as the logical nots of the signals all with relation to the master clock times T2 and T4 which are utilized as inputs to the latch pair 33. Moreover, as before, the latch pair or master slave flip flop 33 is comprised of NAND gates 33*a*–33*h*, gate 33*h* having an output not scan master ($\overline{SCNS}$) which provides a first input to AND gate 34. AND gate 34 has a second input which bypasses the latch pair 33 and is the mechanical control start of scan signal (MCSOS). In this manner, the $\overline{SCNS}$ signal being high and the MCSOS signal being high, at their coincidence, emits an output from the NAND gate 34 of SOS. Since the master slave latch pair 33 acts as a divide by two circuit, when in the draft mode the $\overline{SCNS}$ is high only a portion of the time (i.e., ½) and therefore SOS has only half the frequency of the MCSOS signal. In this manner, the SOS or start of scan signal illustrated in FIG. 3 is only half the frequency of the SOS signal illustrated in FIG. 2 thus omitting every other scan as compared to that in the quality print mode. Moreover, in the quality mode, the scan slave signal $\overline{SCNS}$ is always high, and therefore upon state coincidence of the MCSOS signal with the $\overline{SCNS}$ signal through NAND gate 34, the repetition rate of the SOS signal is necessarily the same as the MCSOS signal. The various inputs to master slave flip flop or latch pair 33 are generated by master slave flip flop 32 (high speed pick slave) HSPS, and by master slave flip flop or latch pair 33 itself, as may be seen from the drawing.

Upon generation of the appropriate start of scan SOS signal, this signal is fed into the drop placement generator logic circuitry 19 which is activated, as set forth heretofore, by the coincidence of a scan ready signal and a start of scan signal SOS which comes from the mechanism control. The scan ready signal, of course, comes from the character generator 13, in a manner which will be explained hereinafter. As shown best in FIG. 7, the drop placement logic circuitry 19 includes an input latch or RS flip flop 35, which is coupled to a second latch or RS flip flop 36, and a third latch or flip flop 37 with appropriate input signals into a latch pair or master slave flip flop 38. Each of the latches or RS flip flops 35–37 is comprised of NAND gates 35*a*–35*d*, 36*a*–36*d*, 37*a*–37*c*, while latch pair 38 is comprised of NAND gates 38*a*–38*h*. A forty-seven bit shift register 40 which is controlled by a shift register control circuit 41 permits the scan data, in a manner to be more fully explained hereinafter, to be unloaded from the shift register 40 through the bus 19*a* to the digital to analog converter or DAC 20 (FIG. 5) and thus to the charge electrode 4 associated with a nozzle 2*a* of the drop generator 2.

The operation of the drop placement logic circuitry 19 is as follows. When the start of scan signal SOS is applied to the latch or RS flip flop circuit 35, if the remainder of the circuitry composed of latch pairs 38, and other latches 37 and 36 have completed their operations from the previous scan, the latches will emit a load shift register signal LDSR (see output LDSR from NAND gate 37*b*) which will permit data (one scan of data) to be transferred from the forty bit shift register 39 in the character generator 13 into the scan data register 40 (47 bit shift register) in the drop placement logic 19. As illustrated, the forty bit shift register 39 in the character generator 13 has a serial data input SDI which receives scan data derived from the font memory. When the shift register 39 is loaded, a signal is outputted to the scan ready decode circuit 39*a* (a multi input AND gate) which with a coincidence of LDSR from gate 37*b* indicates that a scan is ready. As noted, shift register control 41 requires the same load shift register signal LDSR along with the SCAN signal which is outputted from latch pair 38 (NAND gate 38*h*) to permit the scan data

to be unloaded along bus 39*d* to the forty-seven bit shift register 40, and ultimately outputted through the bus 19*a* to the digital to analog converter 20.

The latch or RS flip flop 35 serves to store the SOS or start of scan signal until the previous scan is completed (note the requirement of NAND gates 35*a* and 35*b* for an input of start of scan delayed (SOSD) as well as clock time T2 while that for NAND gate 35*b* of $\overline{\text{SCANST}}$ which is scan stored and clock time T4). The scan stored SCANST signal is outputted from latch pair 38 while SOSD (start of scan delayed) comes from the output of latch 36. In this manner, the circuit illustrated in FIG. 7 allows for exceptionally long serial scan data which may overlap the next adjacent start of scan SOS signal while permitting the scanned data to catch up when the scan data on the preceding and succeeding scans is shorter than the exceptional long serial scan data.

In the IBM 6640 Document Printer, it is necessary to compensate for aerodynamic effects on the ink drops, as well as the induction effect of one charge drop upon the next charge drop so that the drops will be applied to the paper in their proper relative position. The scheme which is employed in the IBM 6640 document printer is described in U.S. Pat. No. 4,086,601 issued on Apr. 25, 1978 to Fillmore, et al. As disclosed in the aforementioned patent, the output of the least significant bits of shift register 40, for example the last 7 bits or even more are used as a memory address to the main memory 42 of a correction data memory, as illustrated in FIG. 8. The output of the main correction memory 42 is applied to a memory data bus which is in turn gated to the DAC 20 via bus 19*a* to apply the scan data voltage that should be applied to the charge electrode for each bit representation of a drop in the scan. The correction data memory includes a pedestal voltage, in the event that a drop is to be printed, so that the drop may clear the gutter. For a more detailed discussion of the guard drop scheme employed, reference to the aforementioned U.S. patent is suggested.

In an alternative mode of operation, instead of "scan skipping", the machine may be set up to provide row skipping or alternate dot skipping during the vertical scan of the drops in forming a letter or a character. (While this alternate embodiment is not the preferred embodiment, with certain character fonts, the visual impression may be more pleasing). For example, every other drop which normally would be printed in the quality mode may be skipped so that the machine, in a like manner as with scan skipping may be speeded up to approximately twice its quality mode speed. In this connection, in FIG. 6, the latch pair 33 is unnecessary and MCSOS may be directly connected to the RS flip flop 35 illustrated in FIG. 7, MCSOS then equalling SOS for signal purposes. However, the basic circuitry remaining in FIG. 6 may be utilized for obtaining a DRAFT SPEED (DS) signal to the drive motor control for increasing the speed of the carrier.

The second modification that is required in order to achieve a horizontal row skip or so that in one scan only every other dot is printed is the application of the draft speed signal to the shift register control 41 (note DRAFT SPEED*) to allow alternate ones of the scan data for bus 39*b* to be loaded into shift register 40, as under control of the load pulse output from the shift register control 41. In this manner, the shift register is loaded only with alternate binary information which indicates that every other drop is a gutter drop. The third alteration that is necessary, and assuming once again the same guard drop scheme that is described in the U.S. Pat. No. 4,086,601, is to select the draft memory 43 to thereby make aerodynamic and induction charge correction to the drops as through the memory data (MEM DATA) bus as an output from the draft memory 43 instead of the main memory 42.

In another alternate embodiment, the draft mode may include both column skipping and row skipping to achieve a character similar to that illustrated in FIG. 4. In this connection, a combination of the two schemes heretofore described is necessary. That is, the MCSOS start of scan information would have to be operative to permit a alternate or scan skip technique, and the DRAFT SPEED* signal level would be applied to both the shift register control 41 and the draft memory 43 for correction data memory select purposes. Additionally, the microprocessor 10 would necessarily indicate to the machine control that the draft mode requested was to be approximately 4 times that of the quality mode speed, which signal may be derived both from the DRAFT SPEED (DS) signal derived in FIG. 6, and a second signal from the microprocessor, which was, for example, ANDED to indicate the faster speed requirement mandated by the microprocessor.

Thus the present invention provides an ink jet printer having the capability of varying its character resolution with a comensurate change in the speed of printing with relative minor modifications to existing machinery.

Although the invention has been described with a certain degree of particularity, it is understood that the present disclosure has been made only by way of example and that numerous changes in the details of construction and the combination and arrangement of parts may be made without departing from the spirit and scope of the invention as hereinafter claimed:

What is claimed is:

1. In a continuous ink jet printer in which a carrier carries a nozzle for emitting a stream of ink drops at a predetermined rate, an ink drop charging electrode and ink drop deflection electrodes, said carrier being driven orthogonally with respect to an ink drop receiving medium by a variable speed drive means to effect printing on said medium of characters by scanning of ink drops thereon, a first predetermined character resolution defined by a predetermined maximum number of picture elements per unit of length and height in a predetermined size character box, an improvement in the speed of formed characters on said ink drop receiving medium while maintaining the predetermined rate of sid ink stream, said improvement comprising: means responsive to an applied input signal for increasing the speed of said variable speed drive means, and other circuit means responsive to said applied input signal to reduce the number of picture elements in said predetermined size character box to thereby produce characters having a second predetermined character resolution.

2. In a continuous ink jet printer in accordance with claim 1 wherein the reduction in the number of picture elements is per unit of length.

3. In a continuous jet ink jet printer in accordance with claim 1 wherein the reduction in the number of picture elements is per unit of height.

4. In a continuous jet ink jet printer in accordance with claim 1 wherein the reduction in the number of picture elements is per unit of length and height.

5. In an ink jet printer having a carrier with a nozzle mounted thereon for emitting a continuous stream of

ink drops towards an ink drop receiving medium at a predetermined rate, charge electrode means associated with said nozzle and stream to charge ink drops in accordance with desired characters to be printed, and deflection electrode means for deflecting the charged ink drops in accordance with a charge thereon whereby characters are formed by scanning said charged droplets onto said ink drop receiving medium;

character generator means for generating data representative of the charge to be placed on selected drops of a single scan, and detection means to indicate the position and direction of movement of said carrier at any point in time and for generating a first signal comprising a start of scan signal indicative of the time of the start of each scan of ink drops in the formation of a character;

alterable speed carrier drive means for moving said carrier in a direction orthogonally to the direction of motion of said scan, ink drop placement circuit means for receiving said data and said first signal and coupled to said charge electrode for charging said ink drops in accordance with said data to form a single character at a first resolution and at a first speed of carrier movement, an improvement in the speed of forming characters on said ink drop receiving medium while maintaining the predetermined rate of said ink stream, said improvement comprising:

first circuit means for receiving a second signal indicating a change in speed of formation and resolution of a character, said first circuit means including means for generating a third signal upon receipt of said second signal for changing the speed of said drive means to a second speed of carrier movement greater than said first speed;

second circuit means responsive to said second signal for causing the ink drops forming a single character to be formed at a second resolution and at a second speed of carrier movement greater than said first speed.

6. In an ink jet printer in accordance with claim 5 wherein said second circuit means includes means responsive to said second signal and said first start of scan signal for regularly or periodically effectively eliminating the start of scan signal.

7. In an ink jet printer in accordance with claim 5 wherein said second circuit means includes means for rendering selected ones of said start of scan signals inoperable.

8. In an ink printer in accordance with claim 6 including means for by-passing said second circuit means to permit application of said first signal to said ink drop placement circuit in the absence of said second signal.

9. In an ink jet printer in accordance with claim 5 including means for applying said third signal to said ink drop placement circuit means, and means for eliminating the data representative of the charge to be placed on alternate drops of a single scan.

10. In an ink jet printer in accordance with claim 9 wherein said means is responsive to said third signal for eliminating the data representative of the change to be placed on alternate drops of a single scan.

11. An ink jet printer comprising in combination:

an ink drop receiving medium;

a nozzle for emitting a continuous stream of ink drops towards said drop receiving medium at a predetermined rate, said nozzle being spaced from said drop receiving medium;

charge electrode means adjacent said nozzle means for charging ink drops in accordance with desired indicia to be placed on said drop receiving medium, and deflection electrode means for deflecting the charged ink drops in accordance with the charge thereon whereby indicia is formed by scanning said charge drops onto said ink drop receiving medium;

carrier means mounting said nozzle means, charge electrode means and deflection electrode means thereon for moving in a path adjacent to said drop receiving medium;

character generator means for generating data representative of the charge to be placed on selected drops of a single scan, and detection means to indicate the position and direction of movement of said carrier at any point in time and for generating a first signal including a start of scan signal indicative of the time of start of each scan of ink drops in the formation of indicia;

carrier drive means for moving said carrier in said path orthogonal to the direction of motion of said scan, ink drop placement circuit means for receiving said data and said first signal and coupled to said charged electrode for charging said ink drops in accordance with said data to form a single indicia at a first resolution and at a first speed of carrier movement;

said first resolution defined by a predetermined maximum number of picture elements per unit of length and height in a predetermined size indicia box;

a second signal, and first circuit means responsive to said second signal for changing the second of said drive means;

and other circuit means responsive to said second signal to reduce the number of picture elements in said predetermined size indicia box to thereby produce indicia having a second predetermined resolution.

12. An ink jet printer in accordance with claim 11 wherein the reduction in the number of picture elements is per unit of height.

13. An ink jet printer in accordance with claim 11 wherein the reduction and number of picture elements is per unit of length.

14. An ink jet printer in accordance with claim 11 wherein the reduction in a number of picture elements is per unit of length and height.

15. An ink jet printer in accordance with claim 11 wherein said first circuit means for receiving a second signal indicating a change in speed of formation and a decrease in the resolution of indicia includes means for generating a third signal upon receipt of said second signal for changing the speed of said drive means to a second speed of carrier movement greater than said first speed.

16. An ink jet printer in accordance with claim 15 wherein said other circuit means includes means for rendering selected ones of said start of scan signals inoperable.

17. An ink jet printer in accordance with claim 15 wherein said other circuit means comprises means responsive to said second signal and said first start of scan signal for regularly or periodically effectively eliminating the start of scan.

18. An ink jet printer in accordance with claim 15 including means for bypassing said other circuit means to permit application of said first signal to said ink drop

placement circuit means in the absence of said second signal.

19. An ink jet printer in accordance with claim 15 including means for applying said third signal to said ink drop placement circuit means, and means for eliminating the data representative of the charge to be placed on alternate drops of a single scan.

20. An ink jet printer in accordance with claim 19 wherein said other circuit means includes means responsive to said second signal and said first signal for eliminating predetermined start of scan signals.

* * * * *

United States Patent [19]

Lomicka, Jr. et al.

[11] **4,326,813**

[45] **Apr. 27, 1982**

[54] **DOT MATRIX CHARACTER PRINTER CONTROL CIRCUITRY FOR VARIABLE PITCH PRINTING**

[75] Inventors: **Roy F. Lomicka, Jr.**, Chelmsford; **Peter N. Heller**, Newton, both of Mass.

[73] Assignee: **Digital Equipment Corporation**, Maynard, Mass.

[21] Appl. No.: **87,681**

[22] Filed: **Oct. 23, 1979**

Related U.S. Application Data

[63] Continuation-in-part of Ser. No. 955,525, Oct. 30, 1978, abandoned.

[51] **Int. Cl.**[3] **B41J 3/12**

[52] **U.S. Cl.** **400/124**; 400/306; 400/322; 400/323

[58] **Field of Search** 400/124, 119, 120, 121, 400/320, 322, 323, 328, 305, 306; 328/151; 101/93.04, 93.05

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,858,703	1/1975	Duley	400/124
3,891,077	6/1975	Sauerbrunn	400/124
3,950,685	4/1976	Kramer	400/124 X
3,970,183	7/1976	Robinson et al.	400/124
3,990,559	11/1976	Martin et al.	400/124
3,999,168	12/1976	Findley et al.	400/306 X
4,005,772	2/1977	Kieffer et al.	400/306 X
4,026,402	5/1977	Byrd	400/124 X
4,050,563	9/1977	Menhennett	400/124
4,079,824	3/1978	Ku	400/124
4,114,750	9/1978	Baeck et al.	400/124 X
4,116,567	9/1978	San Pietro	400/124
4,125,336	11/1978	Chu	400/124
4,169,683	10/1979	Bernardis et al.	400/124
4,169,684	10/1979	Blom	400/124
4,210,404	7/1980	Hanger	400/124
4,213,714	7/1980	Jones et al.	400/306 X
4,220,924	9/1980	Osann	328/151 X
4,255,061	3/1981	Beery	400/124

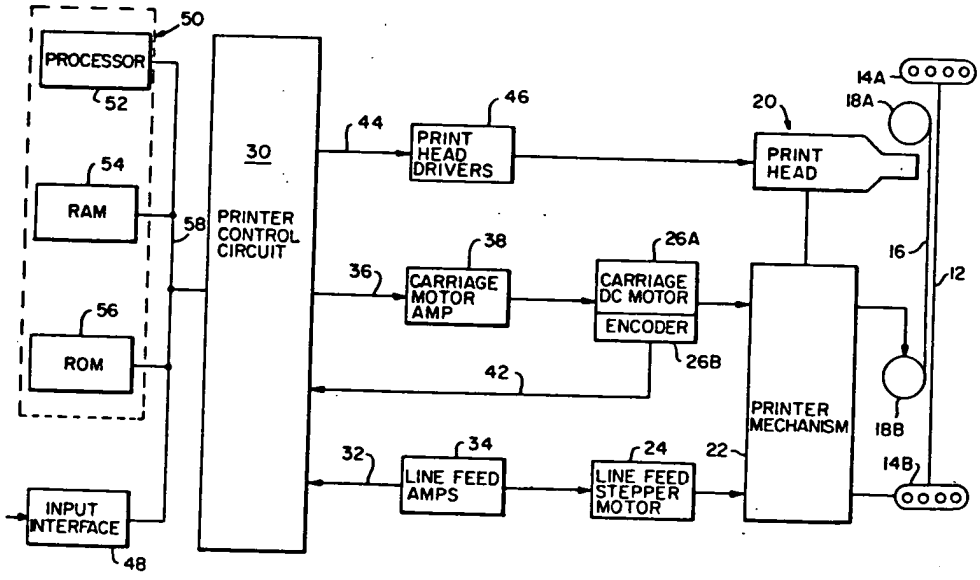
Primary Examiner—Paul T. Sewell

Attorney, Agent, or Firm—Cesari and McKenna

[57] **ABSTRACT**

A dot matrix character printer having the capability of printing characters at variable speeds and with variable pitches. Variable pitch is provided through the use of programmable radix counters which control actuation of the printing solenoids. Solenoid actuation is based on print head position, with the actuation position being varied with print head velocity, to compensate for the distance travelled by the print head during the time between solenoid actuation and print wire impact; this permits operation at variable printing rates. Printing rate is limited by the cyclic rate of the print head, with an additional limitation based on dot printing density which prevents excessive heating of the printing solenoids. Control functions are provided by a firmware-controlled microprocessor and a special purpose integrated circuit (PCC). The operation of the microprocessor and PCC are time-independent of each other. For initiating printing, the microprocessor provides a coarse time window and the PCC controls operations within that window.

22 Claims, 142 Drawing Figures



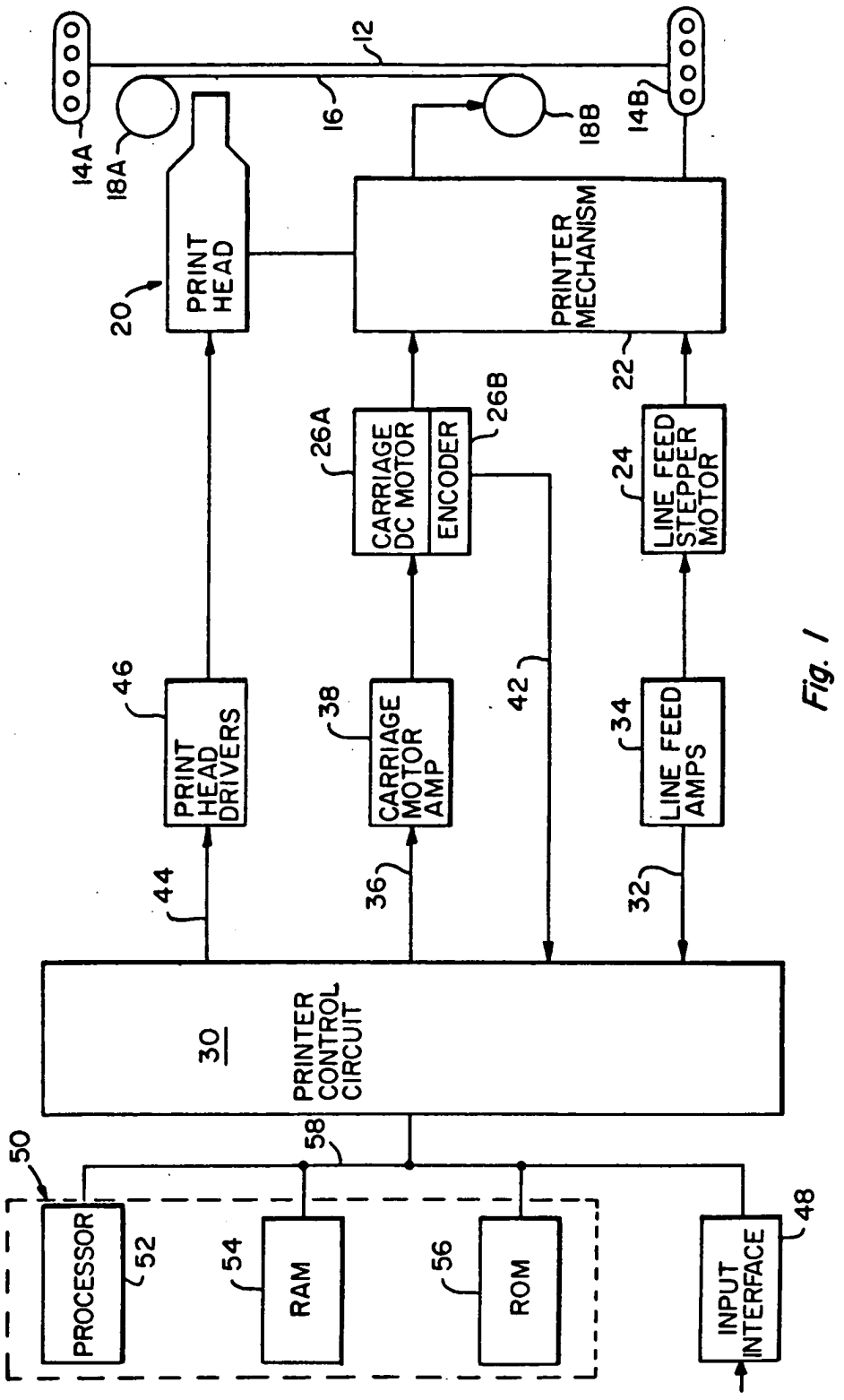

12
16
14A
18A
18B
14B
PRINT HEAD
20
PRINTER MECHANISM
22
CARRIAGE DC MOTOR
26A
ENCODER
26B
LINE FEED STEPPER MOTOR
24
PRINT HEAD DRIVERS
46
CARRIAGE MOTOR AMP
38
42
LINE FEED AMPS
34
44
36
32
30
PRINTER CONTROL CIRCUIT
58
50
PROCESSOR
52
54
RAM
56
ROM
INPUT INTERFACE
48

Fig. 1

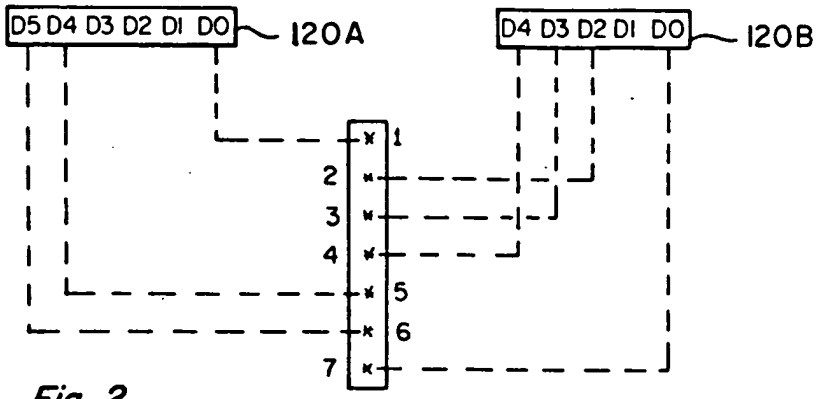


Fig. 2

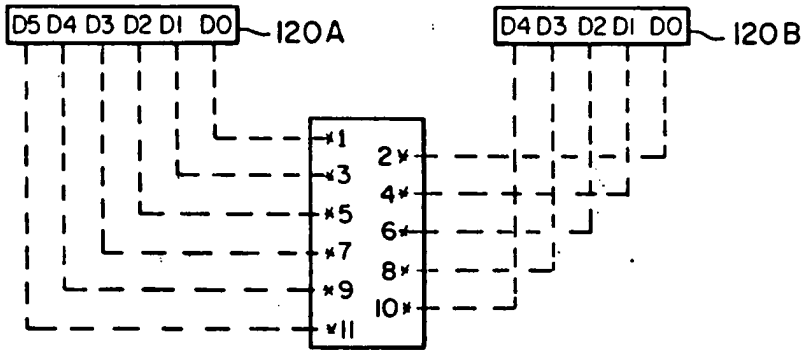


Fig. 3

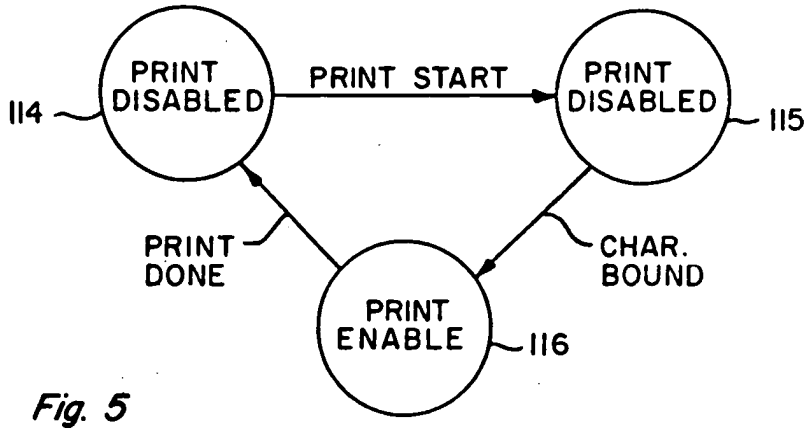


Fig. 5

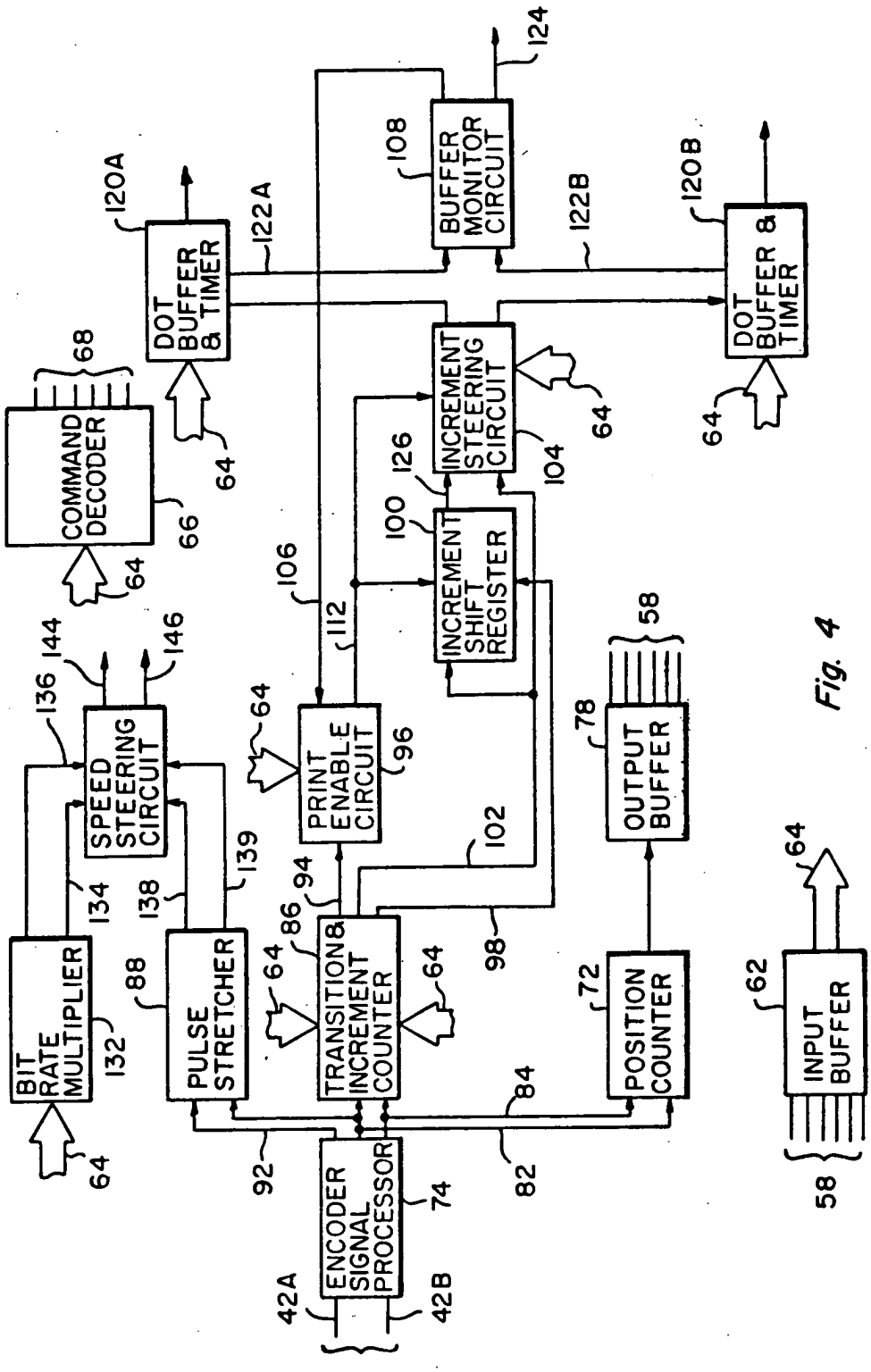

BUFFER MONITOR CIRCUIT
108
124
DOT BUFFER & TIMER
120A
122A
122B
120B
DOT BUFFER & TIMER
INCREMENT STEERING CIRCUIT
104
126
INCREMENT SHIFT REGISTER
100
COMMAND DECODER
68
66
64
106
112
PRINT ENABLE CIRCUIT
96
SPEED STEERING CIRCUIT
136
144
146
139
OUTPUT BUFFER
78
58
102
94
86
98
TRANSITION & INCREMENT COUNTER
PULSE STRETCHER
88
BIT RATE MULTIPLIER
132
134
138
POSITION COUNTER
72
84
INPUT BUFFER
62
ENCODER SIGNAL PROCESSOR
74
92
82
42A
42B

Fig. 4

INPUTS			STATE		OUTPUTS
PRINT START	CHAR. BOUND	PRINT DONE	STATE	NEXT STATE	PRINT ENABLE
0	X	0	114	114	0
1	X	0	114	115	0
X	0	0	115	115	0
X	1	0	115	116	1
X	X	0	116	116	0
X	X	1	X	114	0

Fig. 6

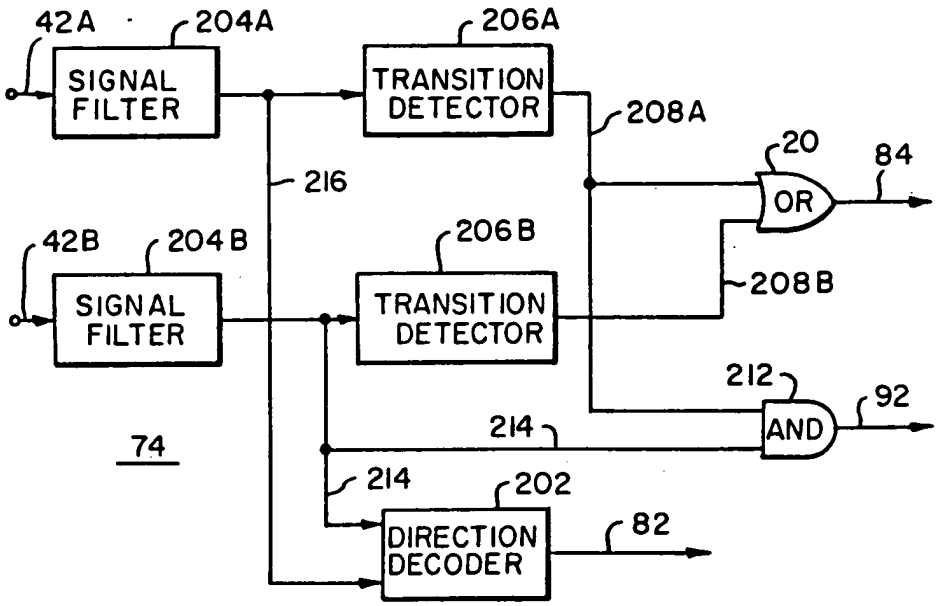


Fig. 7

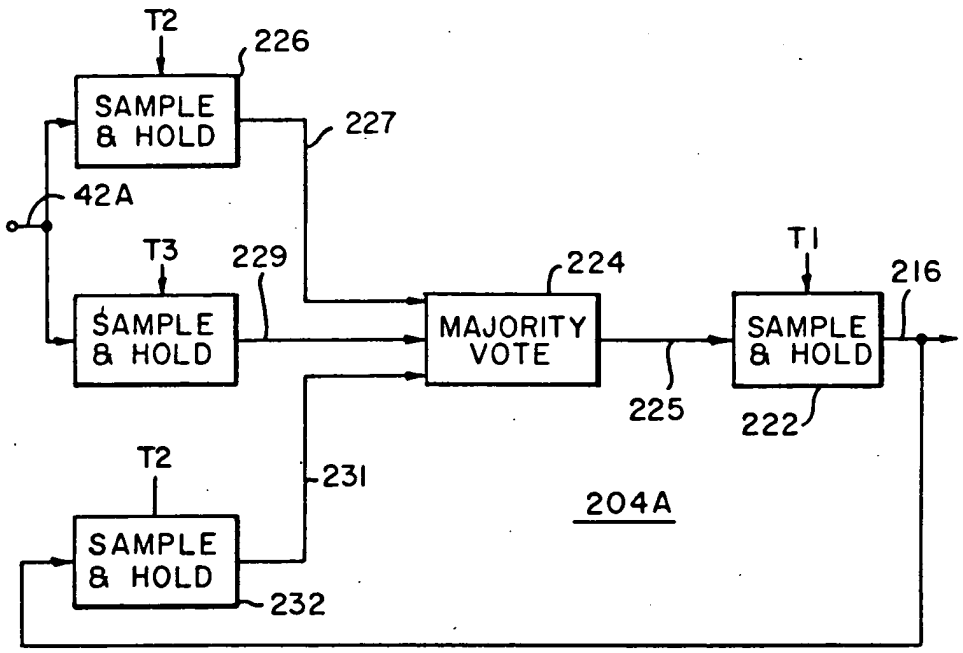


Fig. 8

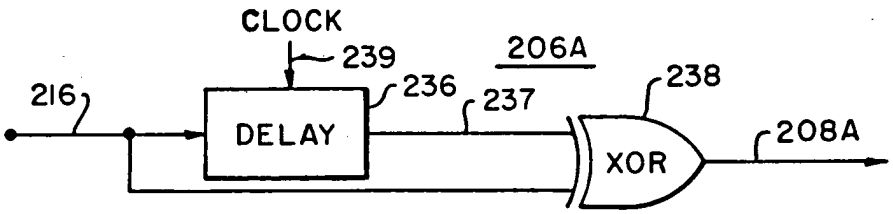


Fig. 9

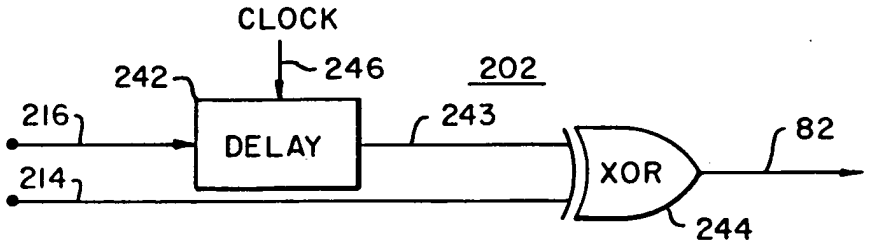


Fig. 10

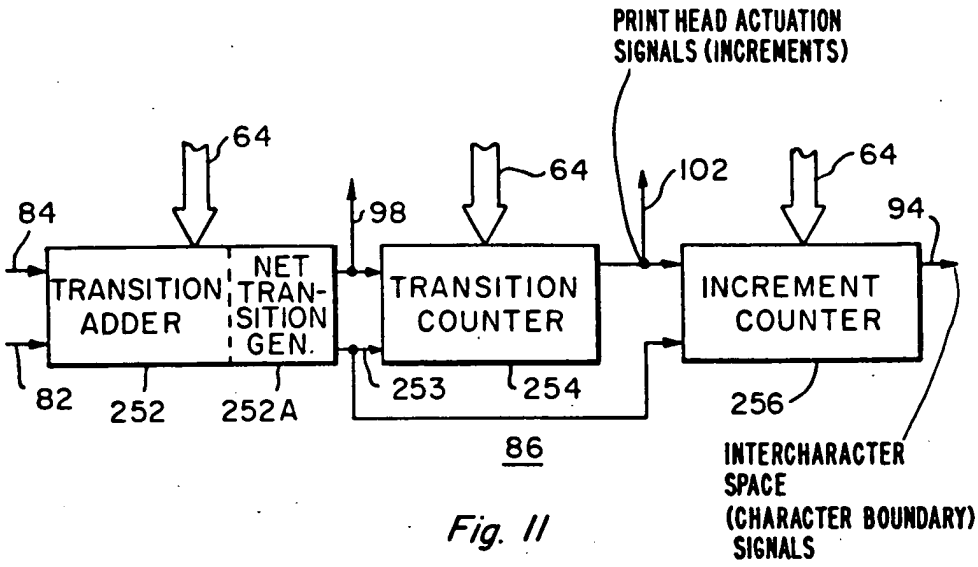

PRINT HEAD ACTUATION SIGNALS (INCREMENTS)
64
64
64
102
84
98
94
TRANSITION ADDER
NET TRAN-SITION GEN.
TRANSITION COUNTER
INCREMENT COUNTER
82
252
252A
253
254
256
86
INTERCHARACTER SPACE (CHARACTER BOUNDARY) SIGNALS

Fig. 11

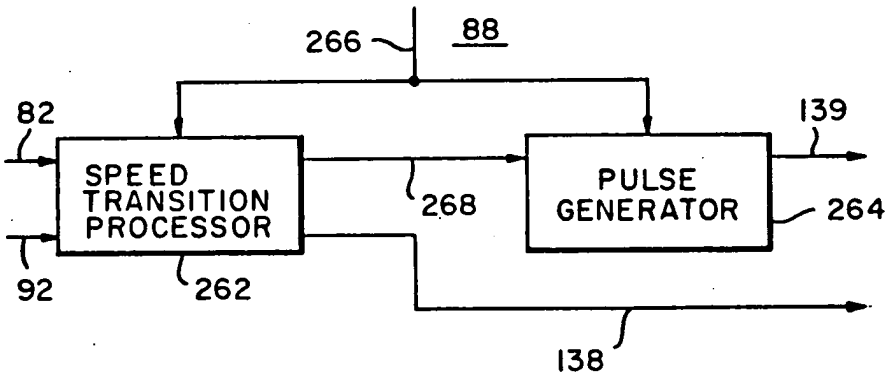

266
88
82
139
SPEED TRANSITION PROCESSOR
268
PULSE GENERATOR
264
92
262
138

Fig. 12

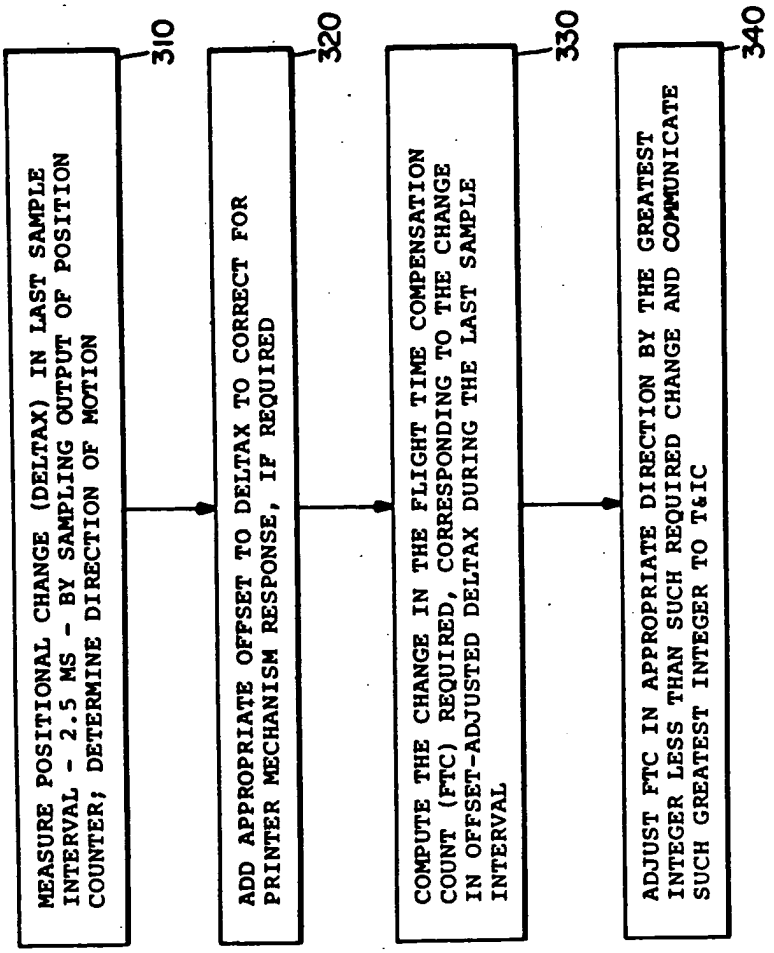

MEASURE POSITIONAL CHANGE (DELTAX) IN LAST SAMPLE INTERVAL - 2.5 MS - BY SAMPLING OUTPUT OF POSITION COUNTER; DETERMINE DIRECTION OF MOTION
310
ADD APPROPRIATE OFFSET TO DELTAX TO CORRECT FOR PRINTER MECHANISM RESPONSE, IF REQUIRED
320
COMPUTE THE CHANGE IN THE FLIGHT TIME COMPENSATION COUNT (FTC) REQUIRED, CORRESPONDING TO THE CHANGE IN OFFSET-ADJUSTED DELTAX DURING THE LAST SAMPLE INTERVAL
330
ADJUST FTC IN APPROPRIATE DIRECTION BY THE GREATEST INTEGER LESS THAN SUCH REQUIRED CHANGE AND COMMUNICATE SUCH GREATEST INTEGER TO T&IC
340

Fig. 13

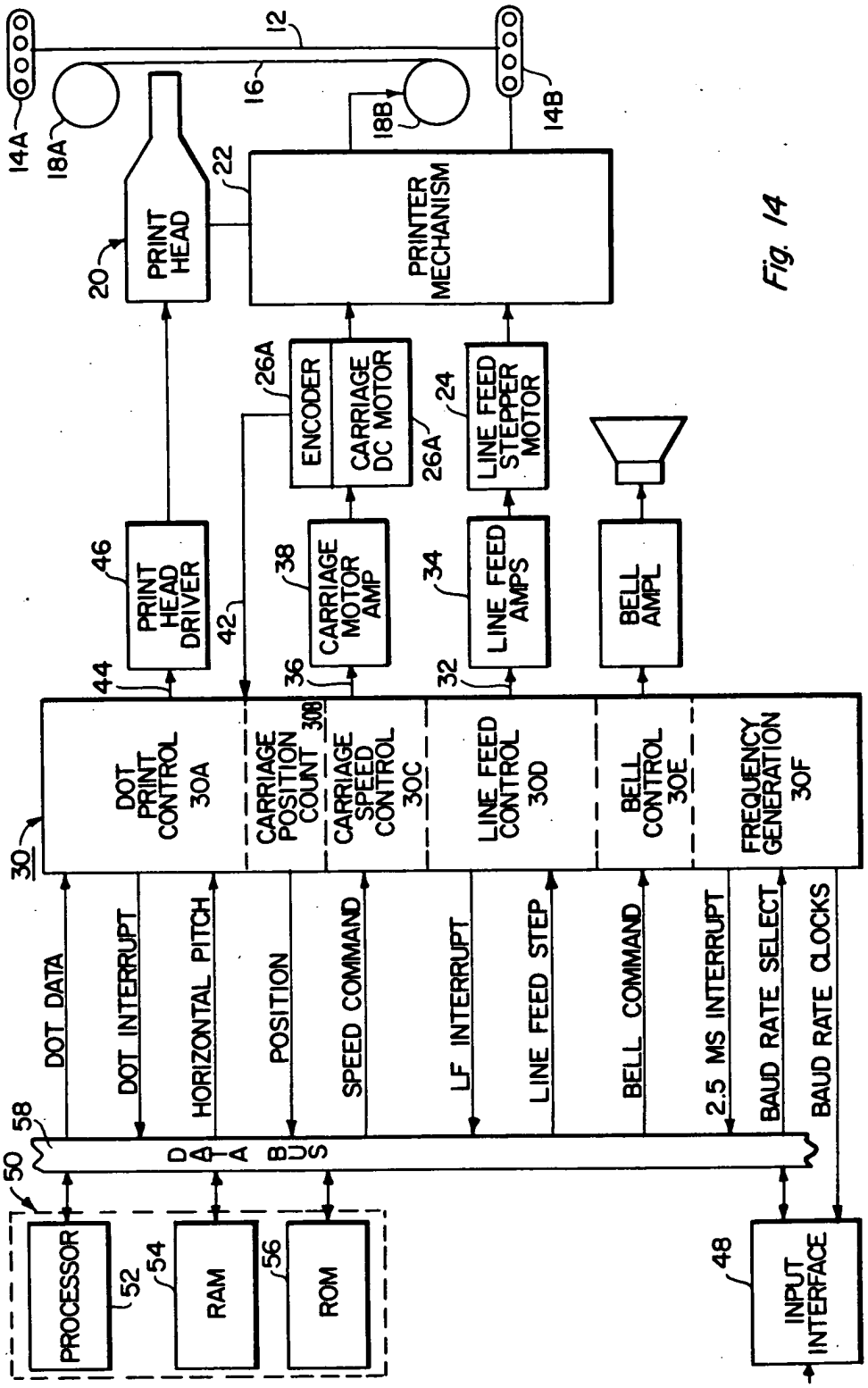

12
16
14A
18A
18B
14B
22
PRINT HEAD
20
PRINTER MECHANISM
26A
ENCODER
CARRIAGE DC MOTOR
24
LINE FEED STEPPER MOTOR
46
PRINT HEAD DRIVER
42
38
CARRIAGE MOTOR AMP
34
LINE FEED AMPS
BELL AMPL
44
36
32
30
DOT PRINT CONTROL 30A
CARRIAGE POSITION COUNT 30B
CARRIAGE SPEED CONTROL 30C
LINE FEED CONTROL 30D
BELL CONTROL 30E
FREQUENCY GENERATION 30F
DOT DATA
DOT INTERRUPT
HORIZONTAL PITCH
POSITION
SPEED COMMAND
LF INTERRUPT
LINE FEED STEP
BELL COMMAND
2.5 MS INTERRUPT
BAUD RATE SELECT
BAUD RATE CLOCKS
58
50
DATA BUS
PROCESSOR
52
54
RAM
56
ROM
48
INPUT INTERFACE

Fig. 14

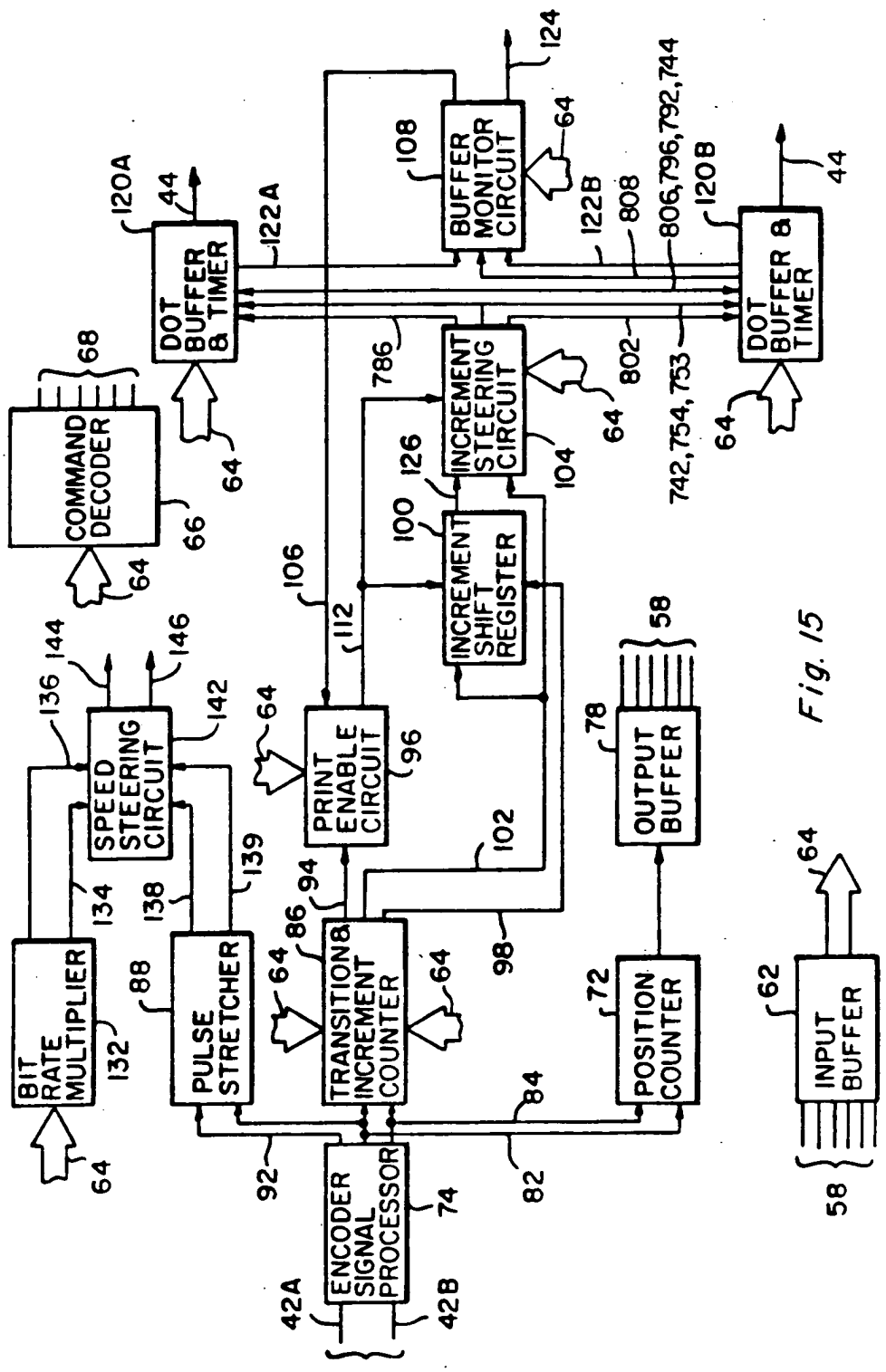

DOT BUFFER & TIMER
BUFFER MONITOR CIRCUIT
DOT BUFFER B TIMER
COMMAND DECODER
INCREMENT STEERING CIRCUIT
INCREMENT SHIFT REGISTER
SPEED STEERING CIRCUIT
PRINT ENABLE CIRCUIT
OUTPUT BUFFER
BIT RATE MULTIPLIER
PULSE STRETCHER
TRANSITION& INCREMENT COUNTER
POSITION COUNTER
INPUT BUFFER
ENCODER SIGNAL PROCESSOR
120A
44
122A
108
124
64
122B
808
806,796,792,744
120B
44
68
786
802
742,754,753
64
66
64
64
126
104
100
106
112
58
136
144
146
142
64
96
78
102
134
138
139
94
86
64
64
98
88
72
132
84
92
82
74
62
64
58
42A
42B

Fig. 15

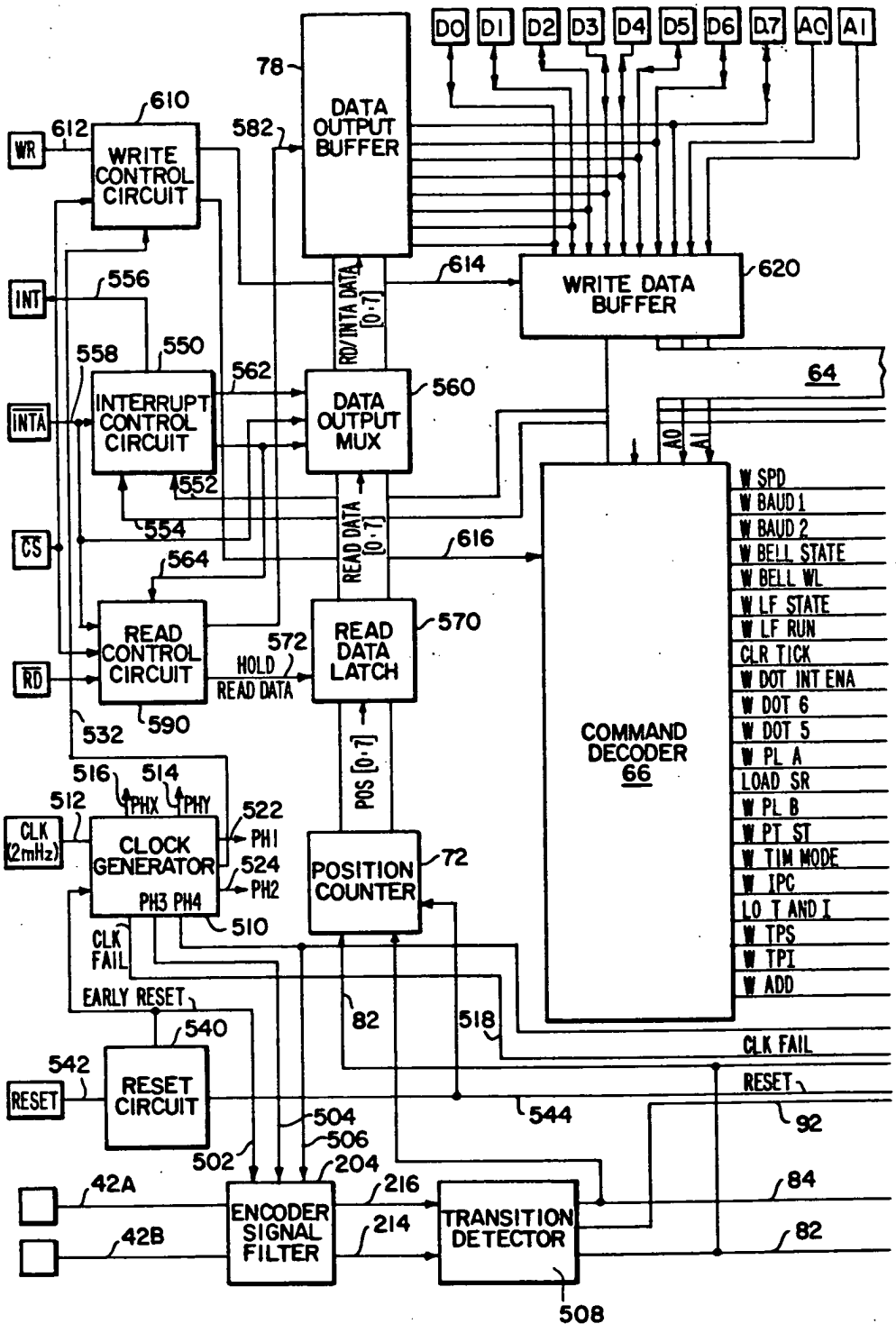

D0
D1
D2
D3
D4
D5
D6
D7
A0
A1
78
DATA OUTPUT BUFFER
610
WR
612
WRITE CONTROL CIRCUIT
582
INT
556
INTA
558
550
INTERRUPT CONTROL CIRCUIT
562
552
554
CS
564
RD
READ CONTROL CIRCUIT
572
HOLD
READ DATA
590
532
DATA OUTPUT MUX
560
RD/INTA DATA [0-7]
READ DATA [0-7]
614
WRITE DATA BUFFER
620
64
616
READ DATA LATCH
570
POS [0-7]
516
514
512
PHX
PHY
CLK (2mHz)
CLOCK GENERATOR
PH3 PH4
522
PH1
524
PH2
510
POSITION COUNTER
72
CLK FAIL
EARLY RESET
82
518
540
542
RESET
RESET CIRCUIT
504
506
502
204
216
214
42A
42B
ENCODER SIGNAL FILTER
TRANSITION DETECTOR
508
84
82
92
544
COMMAND DECODER 66
W SPD
W BAUD 1
W BAUD 2
W BELL STATE
W BELL WL
W LF STATE
W LF RUN
CLR TICK
W DOT INT ENA
W DOT 6
W DOT 5
W PL A
LOAD SR
W PL B
W PT ST
W TIM MODE
W IPC
LO T AND I
W TPS
W TPI
W ADD
CLK FAIL
RESET

Fig. 16A

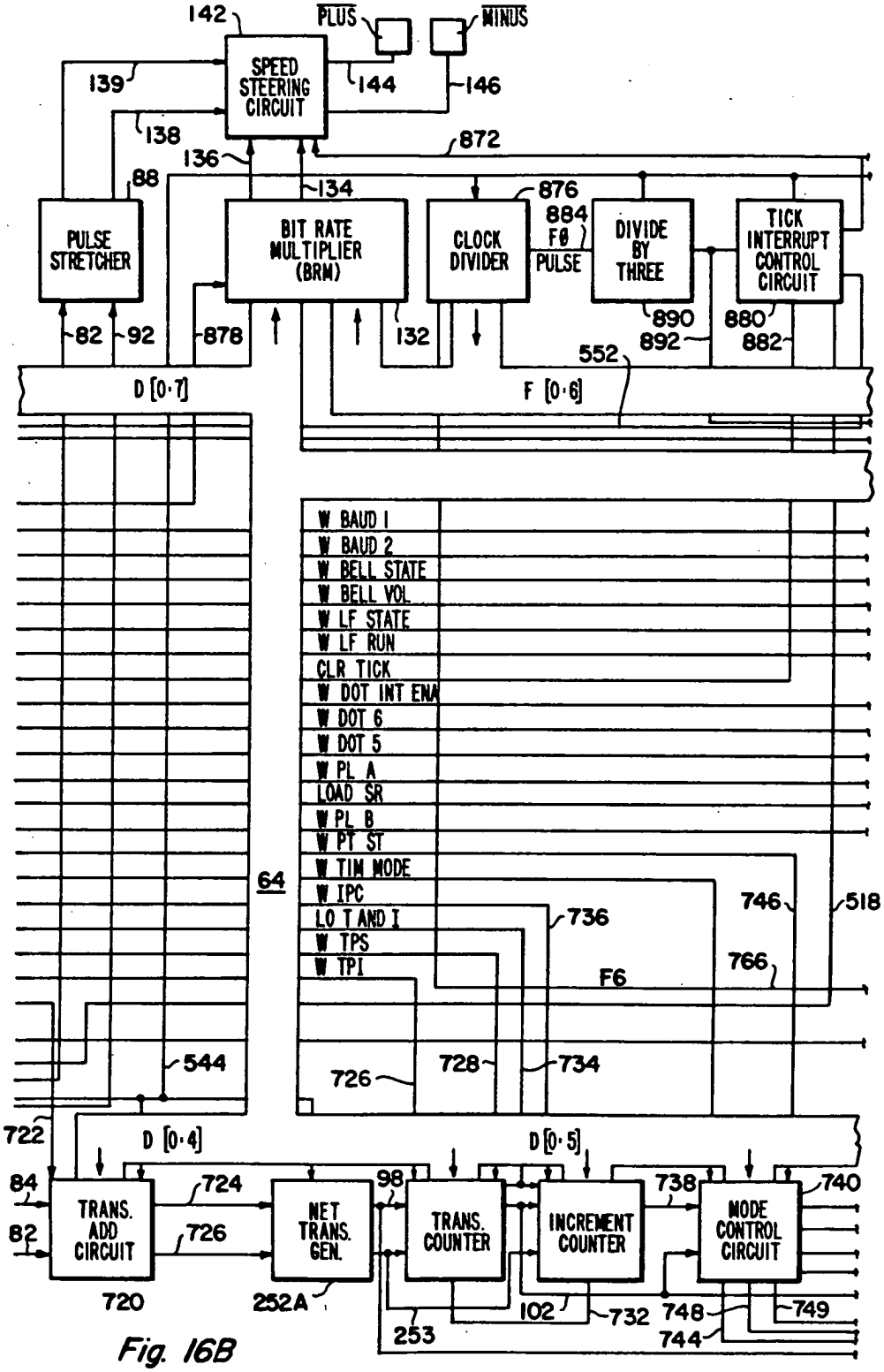

142
PLUS
MINUS
SPEED STEERING CIRCUIT
139
144
146
138
872
136
88
134
876
884
F0
PULSE
PULSE STRETCHER
BIT RATE MULTIPLIER (BRM)
CLOCK DIVIDER
DIVIDE BY THREE
TICK INTERRUPT CONTROL CIRCUIT
82
92
878
132
890
892
880
882
552
D [0·7]
F [0·6]
W BAUD 1
W BAUD 2
W BELL STATE
W BELL VOL
W LF STATE
W LF RUN
CLR TICK
W DOT INT ENA
W DOT 6
W DOT 5
W PL A
LOAD SR
W PL B
W PT ST
W TIM MODE
W IPC
LO T AND I
W TPS
W TPI
64
746
518
736
766
F6
544
726
728
734
722
D [0·4]
D [0·5]
84
82
724
726
98
738
740
TRANS. ADD CIRCUIT
NET TRANS. GEN.
TRANS. COUNTER
INCREMENT COUNTER
MODE CONTROL CIRCUIT
720
252A
102
732
748
749
253
744

Fig. 16B

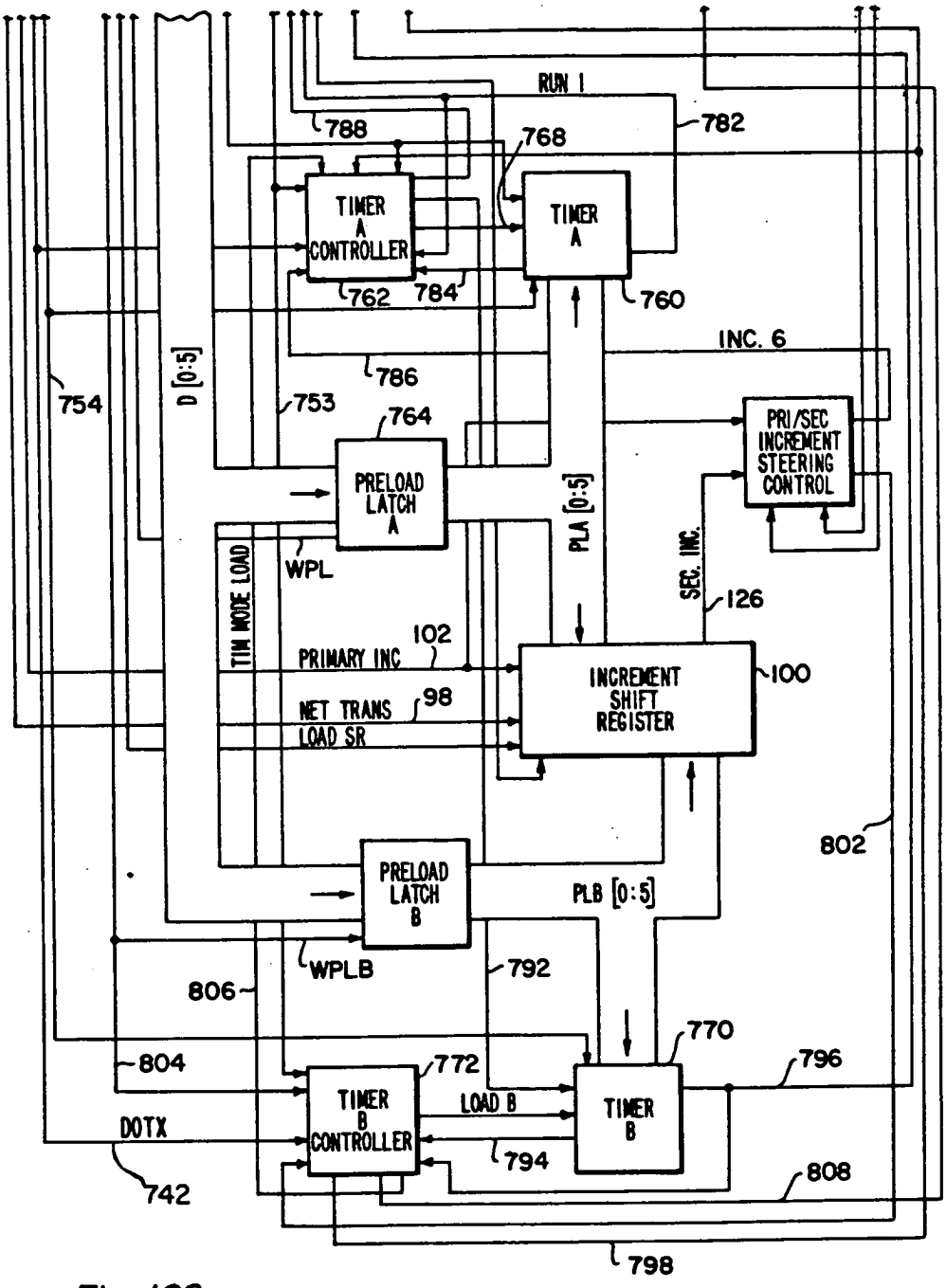

RUN I
788
768
782
TIMER A CONTROLLER
TIMER A
762
784
760
INC. 6
754
D [0:5]
786
753
764
PRI/SEC INCREMENT STEERING CONTROL
PRELOAD LATCH A
PLA [0:5]
WPL
TIM MODE LOAD
SEC. INC.
126
102
PRIMARY INC
INCREMENT SHIFT REGISTER
100
98
NET TRANS
LOAD SR
802
PRELOAD LATCH B
PLB [0:5]
WPLB
792
806
770
804
772
796
TIMER B CONTROLLER
LOAD B
TIMER B
DOTX
794
808
742
798

Fig. 16C

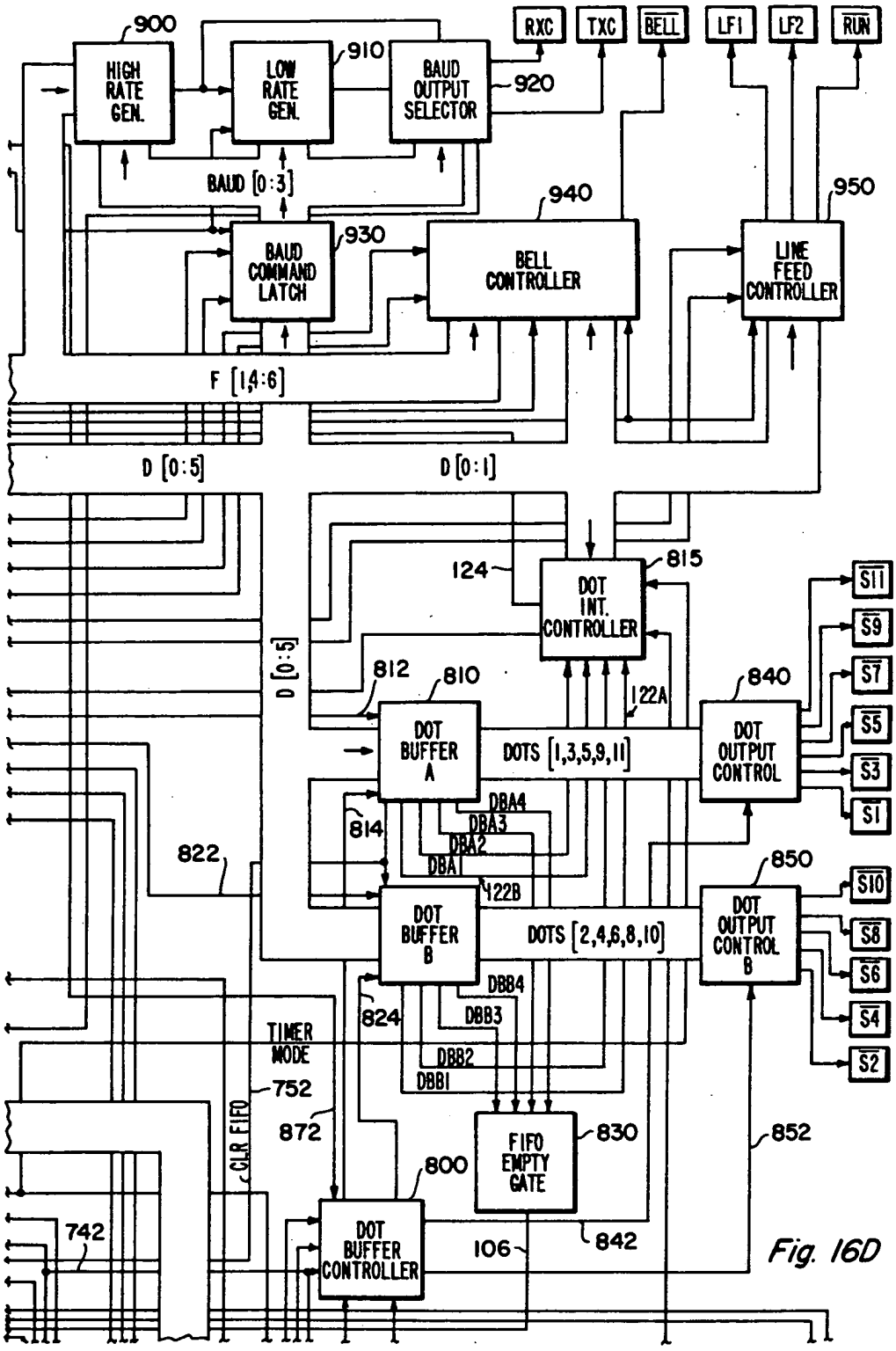

900
HIGH RATE GEN.
910
LOW RATE GEN.
BAUD OUTPUT SELECTOR
920
RXC
TXC
BELL
LF1
LF2
RUN
BAUD [0:3]
940
950
930
BAUD COMMAND LATCH
BELL CONTROLLER
LINE FEED CONTROLLER
F [1,4:6]
D [0:5]
D [0:1]
124
815
DOT INT. CONTROLLER
S11
S9
S7
S5
S3
S1
812
810
840
122A
DOT BUFFER A
DOTS [1,3,5,9,11]
DOT OUTPUT CONTROL
D [0:5]
DBA4
DBA3
DBA2
DBA1
814
822
850
122B
DOT BUFFER B
DOTS [2,4,6,8,10]
DOT OUTPUT CONTROL B
S10
S8
S6
S4
S2
DBB4
DBB3
DBB2
DBB1
824
TIMER MODE
752
CLR FIFO
872
FIFO EMPTY GATE
830
852
800
742
DOT BUFFER CONTROLLER
106
842

Fig. 16D

Fig. 17A

APPARATUS BLOCK NAME	NO.	FIG. NO.	CORRESPONDING CIRCUIT FIG. NO.
ENCODER SIGNAL FILTER	204	16A	18X
TRANSITION DETECTOR	508	16A	18Y
RESET CIRCUIT	540	16A	18A
CLOCK GENERATOR	510	16A	18B
POSITION COUNTER	72	16A	18J
READ DATA LATCH	570	16A	18G
DATA OUTPUT MULTIPLEXER	560	16A	18F
DATA OUTPUT BUFFER	78	16A	18E
READ CONTROL CIRCUIT	590	16A	18D
INTERRUPT CONTROL CIRCUIT	550	16A	18C
WRITE CONTROL CIRCUIT	610	16A	18I
WRITE DATA BUFFER	620	16A	18H
COMMAND DECODER	66	16A	18K
TRANSITION ADD CIRCUIT	720	16B	18Z
NET TRANSITION GENERATOR	252A	16B	18AA
TRANSITION COUNTER	254	16B	18AB(1),(2)
INCREMENT COUNTER	256	16B	18AC

Fig. 17B

APPARATUS BLOCK			CORRESPONDING CIRCUIT
NAME	NO.	FIG. NO.	FIG. NO.
PULSE STRETCHER	88	16B	18M
SPEED STEERING CIRCUIT	142	16B	18N
BIT RATE MULTIPLEXER (BRM)	132	16B	18L
CLOCK DIVIDER	876	16B	18Q
TICK INTERRUPT CONTROL	880	16B	18(O)
DIVIDE-BY-3	890	16B	18P
MODE CONTROL CIRCUIT	740	16C	18AD
TIMER A CONTROLLER	762	16C	18AI
TIMER A	760	16C	18AG
PRELOAD LATCH A	764	16C	18AE
TIMER B CONTROLLER	772	16C	18AJ-1
TIMER B	770	16C	18AH
PRELOAD LATCH B	774	16C	18AF
INCREMENT SHIFT REGISTER	100	16C	18AJ-2
PRIMARY/SECONDARY INCREMENT STEERING CONTROL	750	16C	18AM
DOT BUFFER CONTROLLER	800	16D	18AN
DOT BUFFER A	810	16D	18AK
DOT BUFFER B	820	16D	18AL
DOT INTERRUPT CONTROL CIRCUIT	815	16D	18AP
FIFO EMPTY GATE	830	16D	18AO

Fig. 17C

APPARATUS BLOCK NAME	NO.	FIG. NO.	CORRESPONDING CIRCUIT FIG. NO.
DOT OUTPUT CONTROL CIRCUIT A	840	16D	18AQ
DOT OUTPUT CONTROL CIRCUIT B	850	16D	18AR
BAND COMMAND LATCH	930	16D	18R
HIGH RATE GEN.	900	16D	18S
BELL CONTROLLER	940	16D	18T
BAND OUTPUT SEL	920	16D	18U
LOW RATE GENERATOR	910	16D	18V
LINE FEED CONTROLLER	950	16D	18W

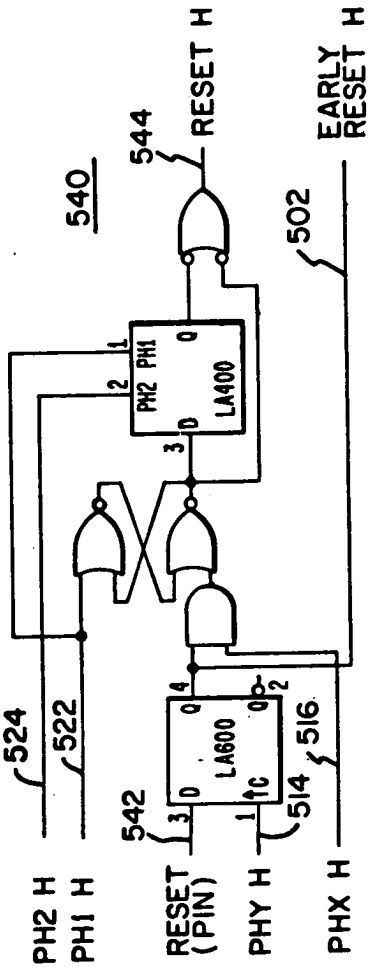


Fig. 18A

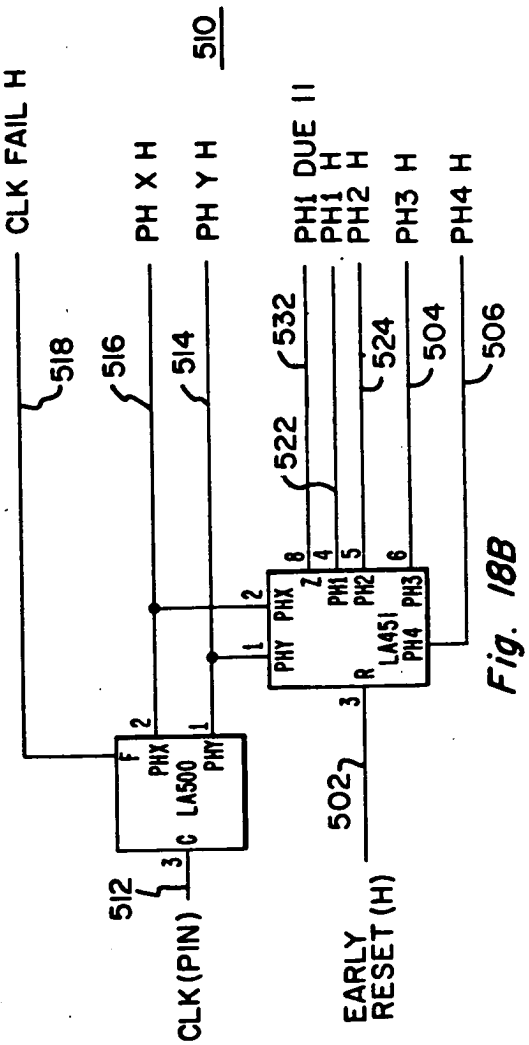


Fig. 18B

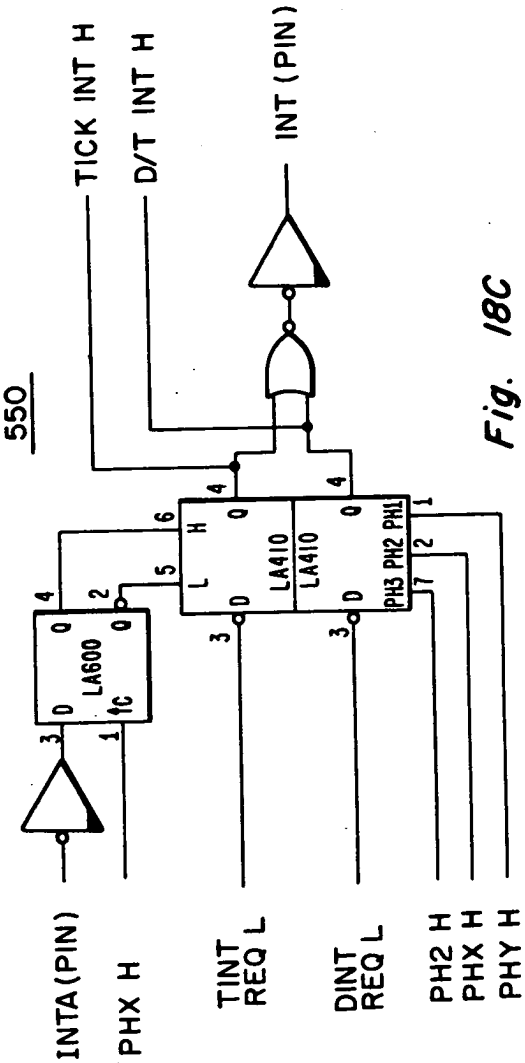


Fig. 18C

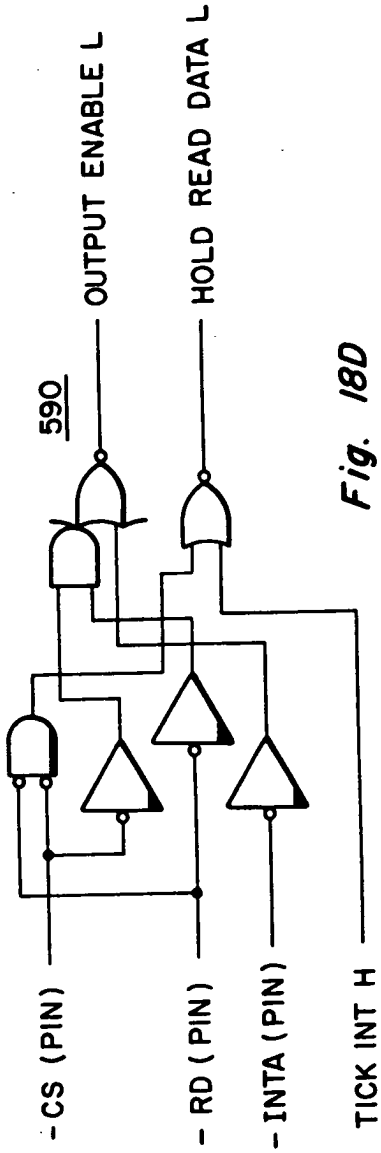


Fig. 18D

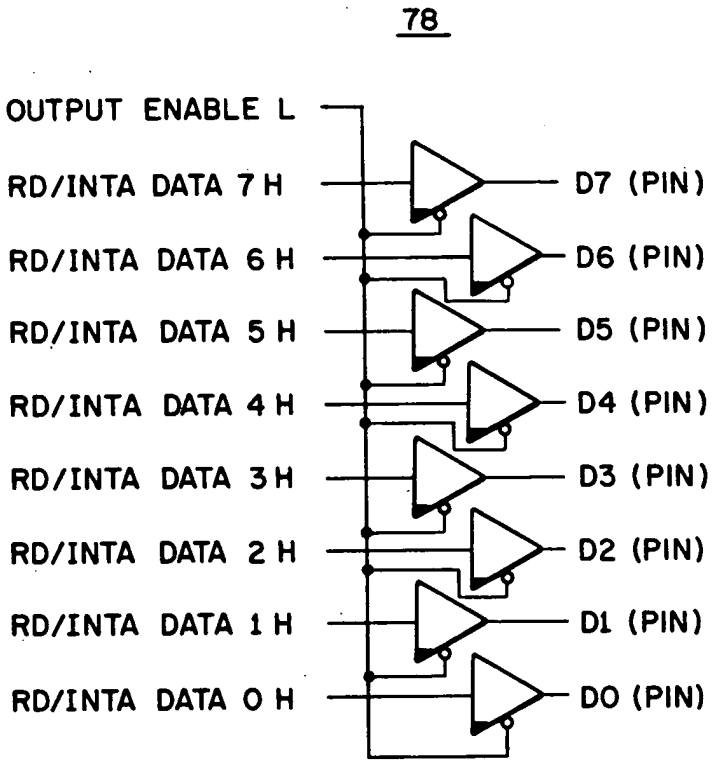

78
OUTPUT ENABLE L
RD/INTA DATA 7 H
RD/INTA DATA 6 H
RD/INTA DATA 5 H
RD/INTA DATA 4 H
RD/INTA DATA 3 H
RD/INTA DATA 2 H
RD/INTA DATA 1 H
RD/INTA DATA 0 H
D7 (PIN)
D6 (PIN)
D5 (PIN)
D4 (PIN)
D3 (PIN)
D2 (PIN)
D1 (PIN)
DO (PIN)

Fig. 18E

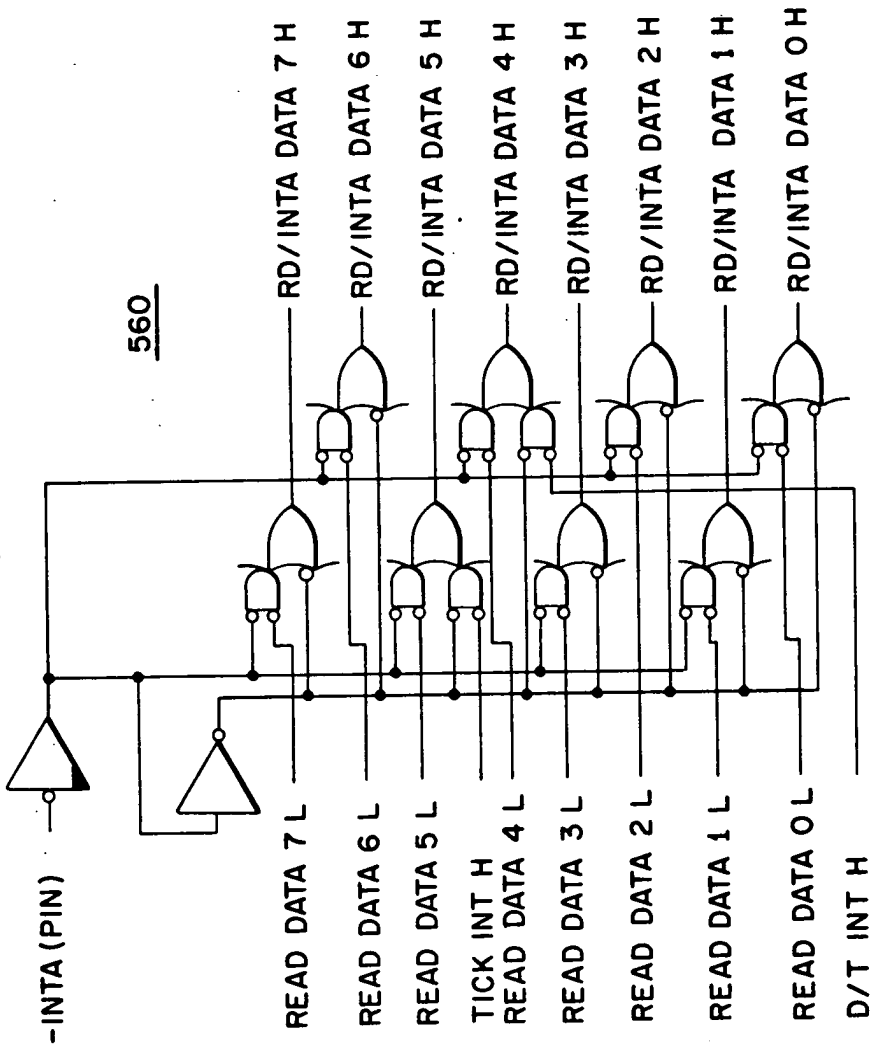

560
RD/INTA DATA 7 H
RD/INTA DATA 6 H
RD/INTA DATA 5 H
RD/INTA DATA 4 H
RD/INTA DATA 3 H
RD/INTA DATA 2 H
RD/INTA DATA 1 H
RD/INTA DATA 0 H
-INTA (PIN)
READ DATA 7 L
READ DATA 6 L
READ DATA 5 L
TICK INT H
READ DATA 4 L
READ DATA 3 L
READ DATA 2 L
READ DATA 1 L
READ DATA 0 L
D/T INT H

Fig. 18F

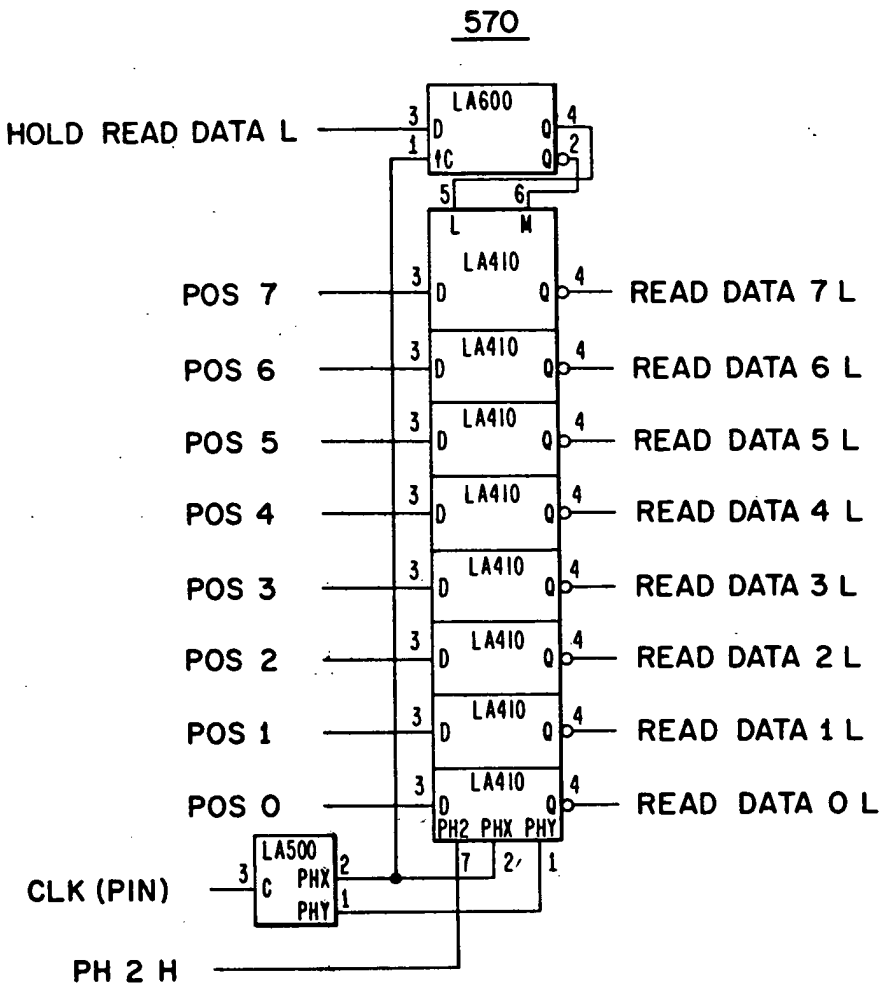

570
HOLD READ DATA L
LA600
D
C
Q
Q
L
M
LA410
POS 7
READ DATA 7 L
POS 6
READ DATA 6 L
POS 5
READ DATA 5 L
POS 4
READ DATA 4 L
POS 3
READ DATA 3 L
POS 2
READ DATA 2 L
POS 1
READ DATA 1 L
POS 0
READ DATA 0 L
PH2 PHX PHY
LA500
C
PHX
PHY
CLK (PIN)
PH 2 H

Fig. 18G

620

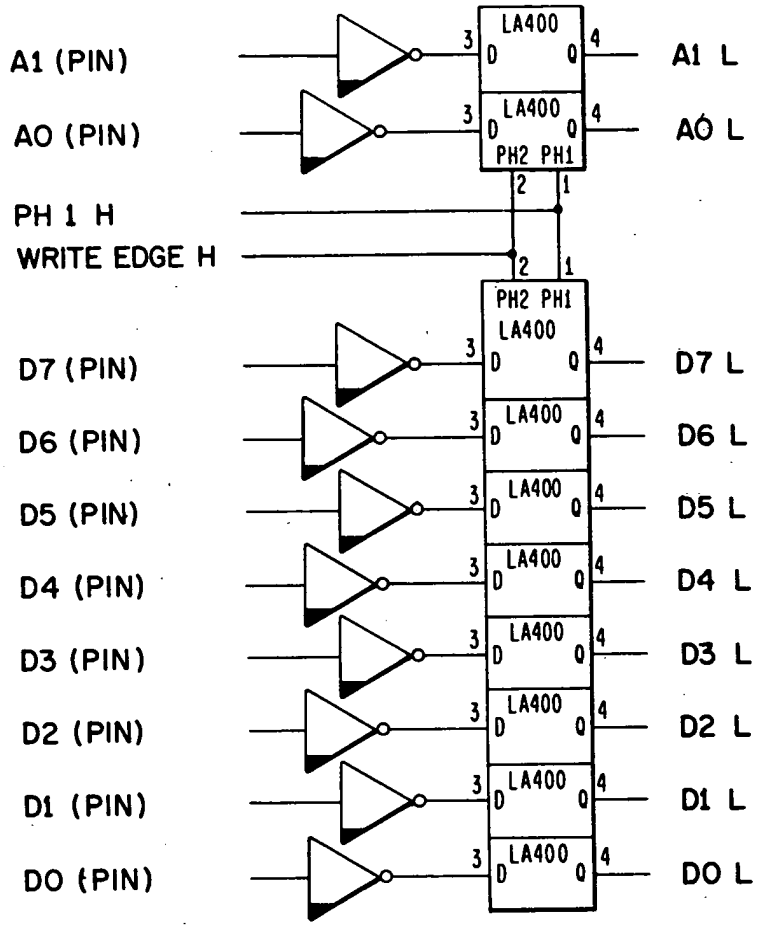

A1 (PIN)
AO (PIN)
PH 1 H
WRITE EDGE H
D7 (PIN)
D6 (PIN)
D5 (PIN)
D4 (PIN)
D3 (PIN)
D2 (PIN)
D1 (PIN)
DO (PIN)
LA400
D
Q
PH2 PH1
A1 L
AO L
D7 L
D6 L
D5 L
D4 L
D3 L
D2 L
D1 L
DO L

Fig. 18H

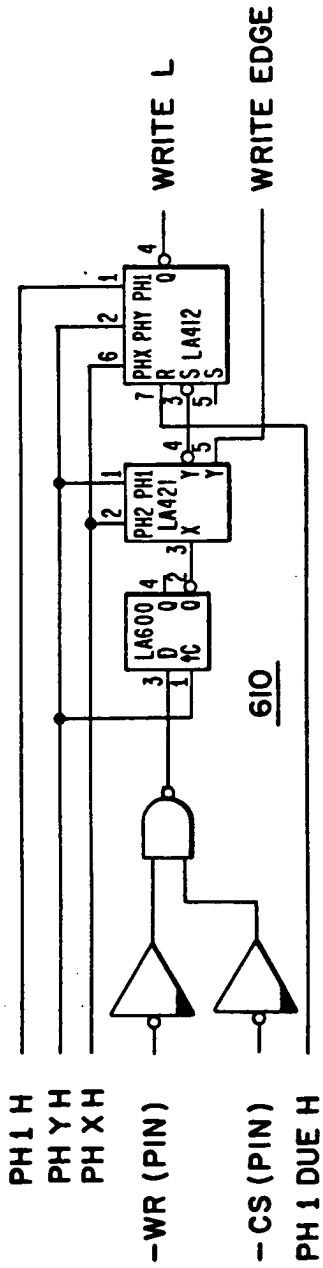


Fig. 18I

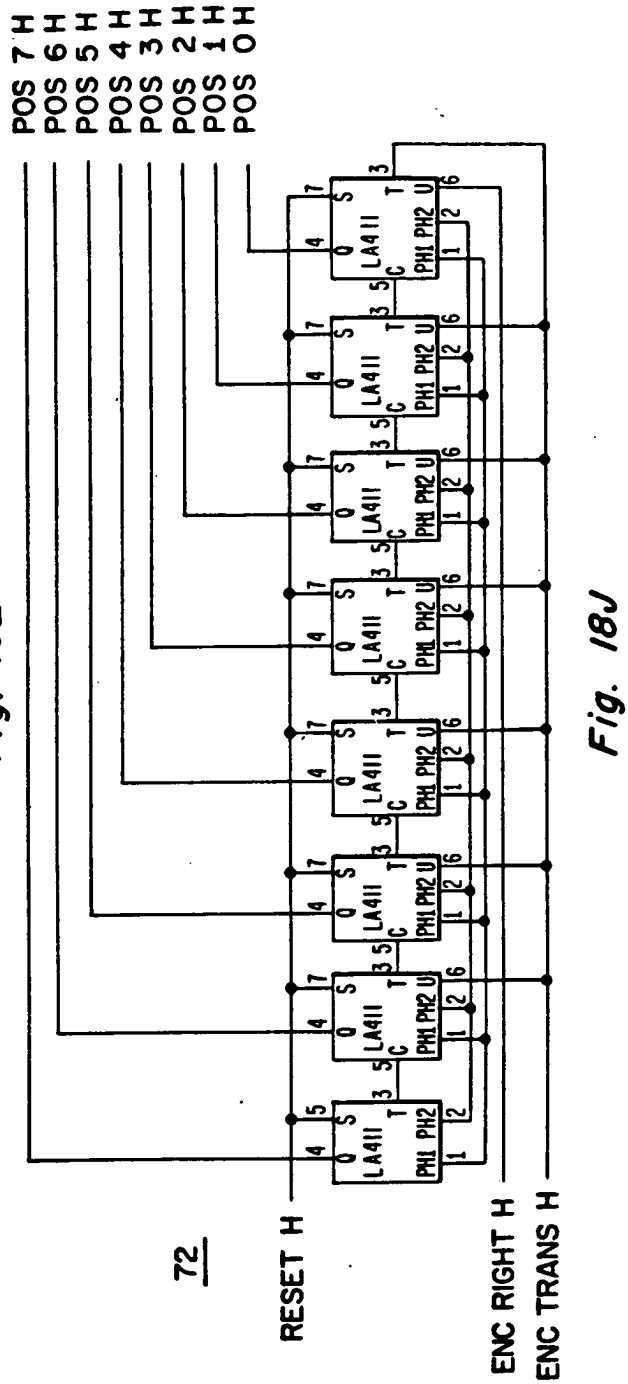


Fig. 18J

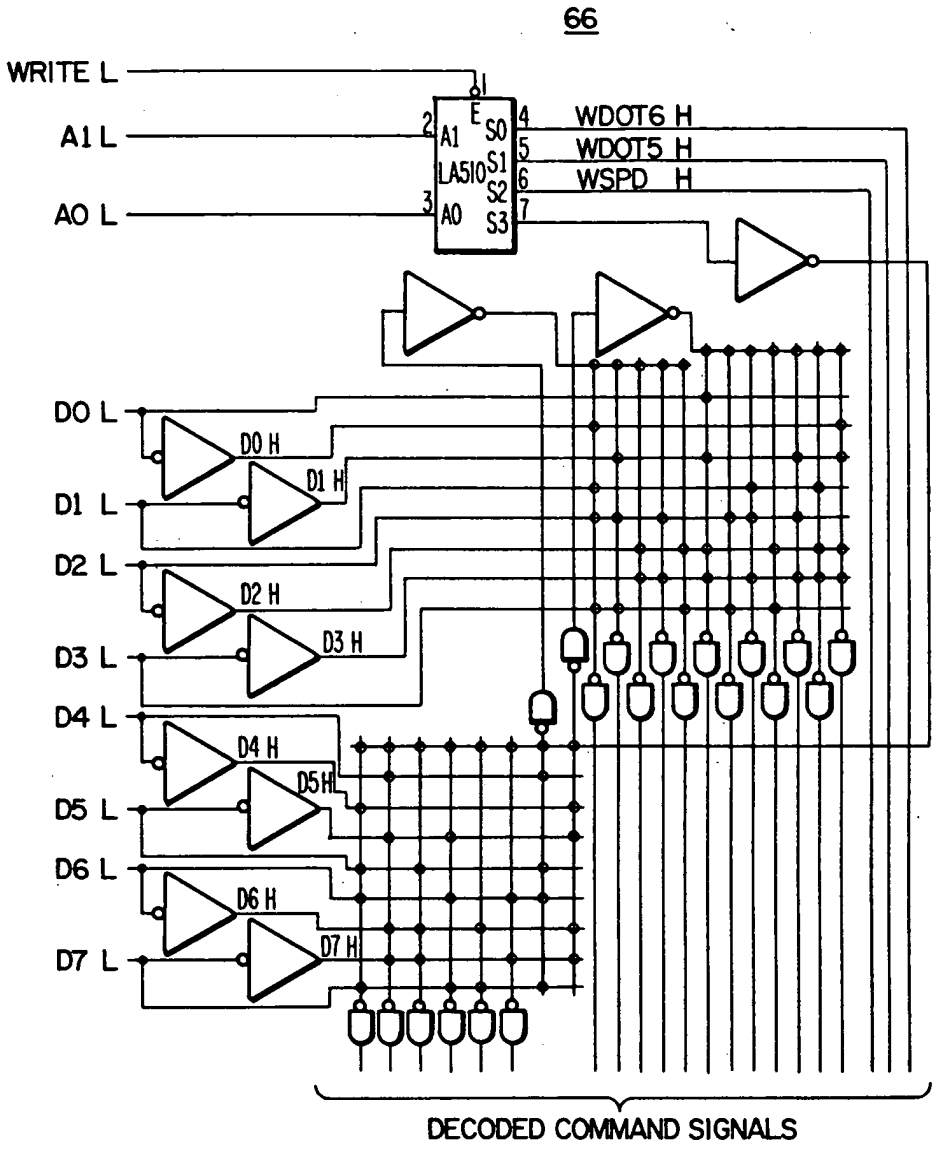

66
WRITE L
A1 L
AO L
LA510
E
A1
AO
S0
S1
S2
S3
WDOT6 H
WDOT5 H
WSPD H
DO L
D1 L
D2 L
D3 L
D4 L
D5 L
D6 L
D7 L
DO H
D1 H
D2 H
D3 H
D4 H
D5 H
D6 H
D7 H
DECODED COMMAND SIGNALS

Fig. 18K

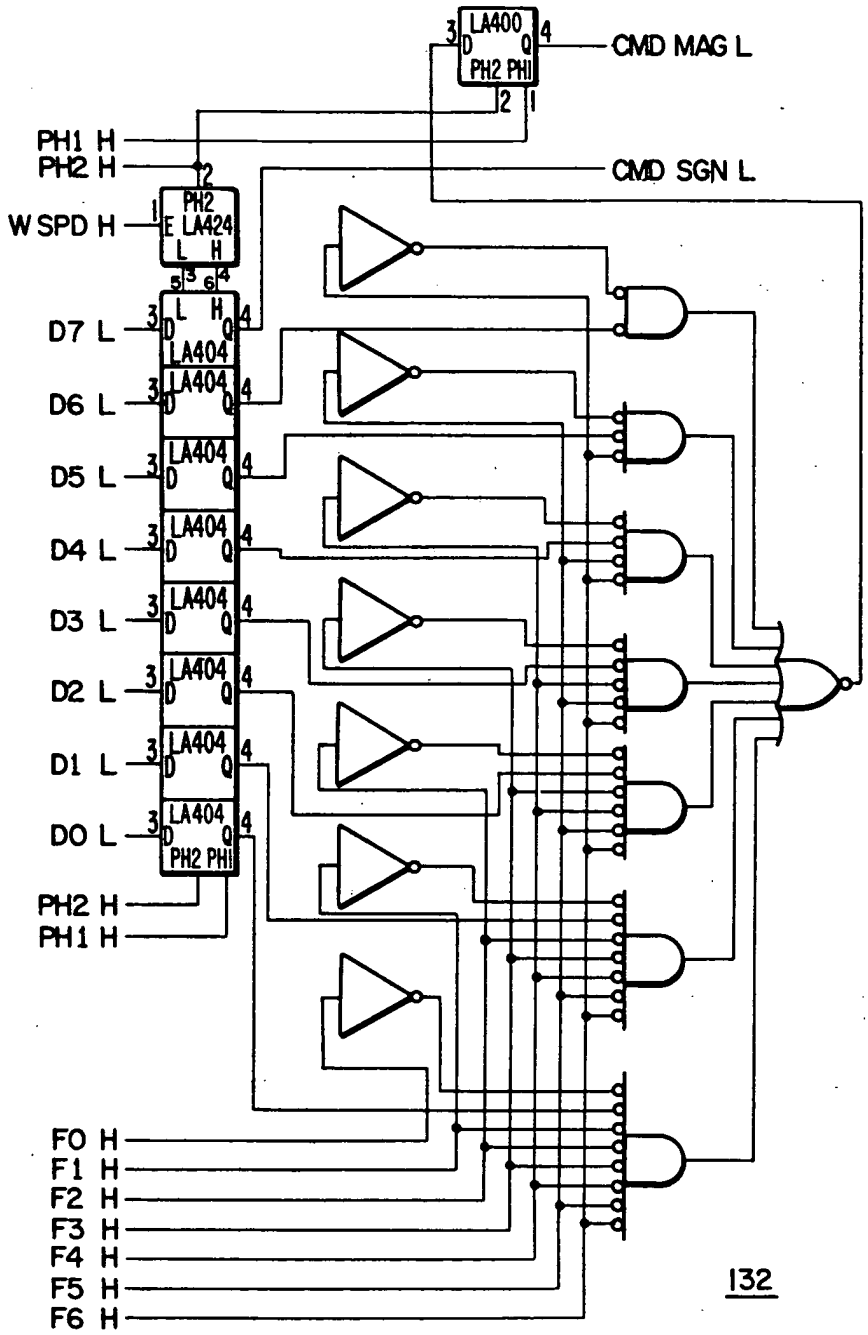

LA400
PH2 PH1
CMD MAG L
PH1 H
PH2 H
CMD SGN L
W SPD H
PH2
E LA424
L H
D7 L
D6 L
D5 L
D4 L
D3 L
D2 L
D1 L
DO L
LA404
PH2 PH1
PH2 H
PH1 H
FO H
F1 H
F2 H
F3 H
F4 H
F5 H
F6 H
132

Fig. 18L

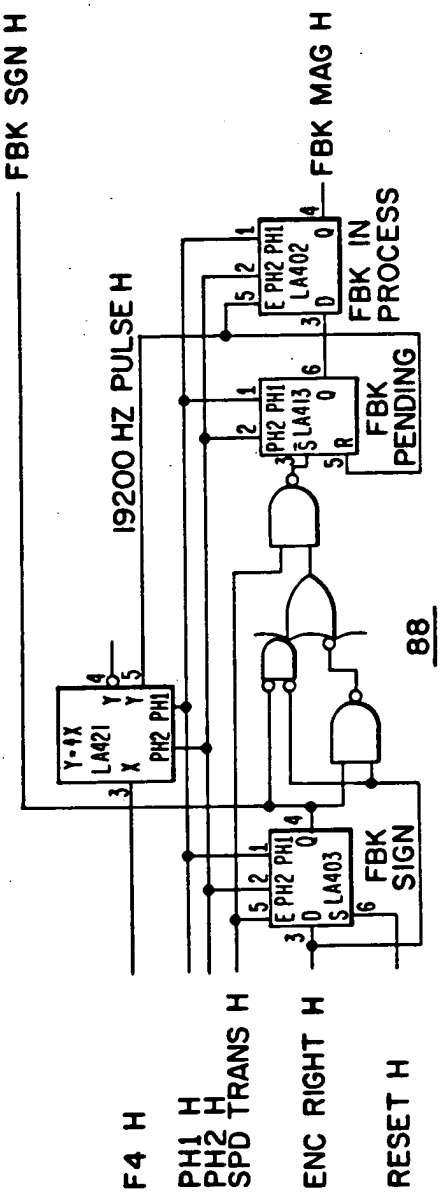


Fig. 18M

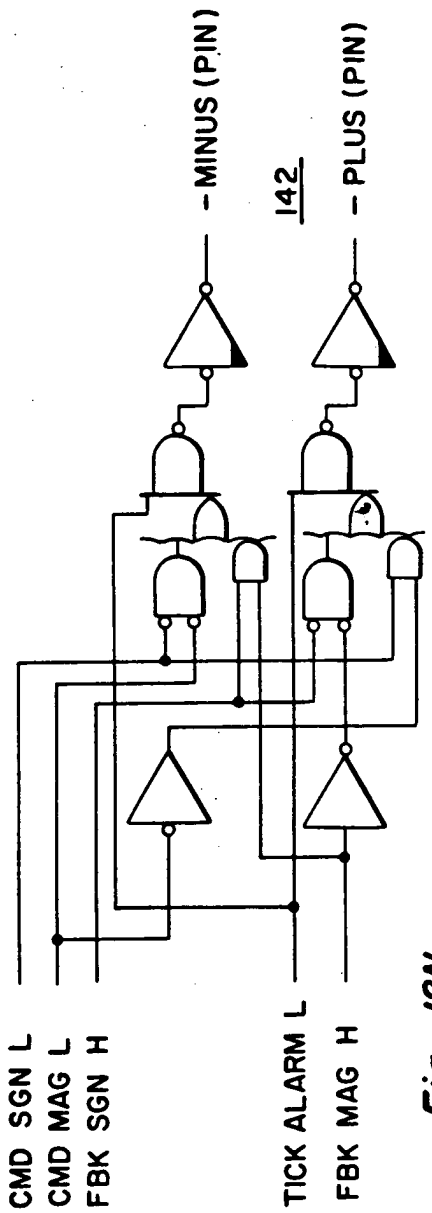


Fig. 18N

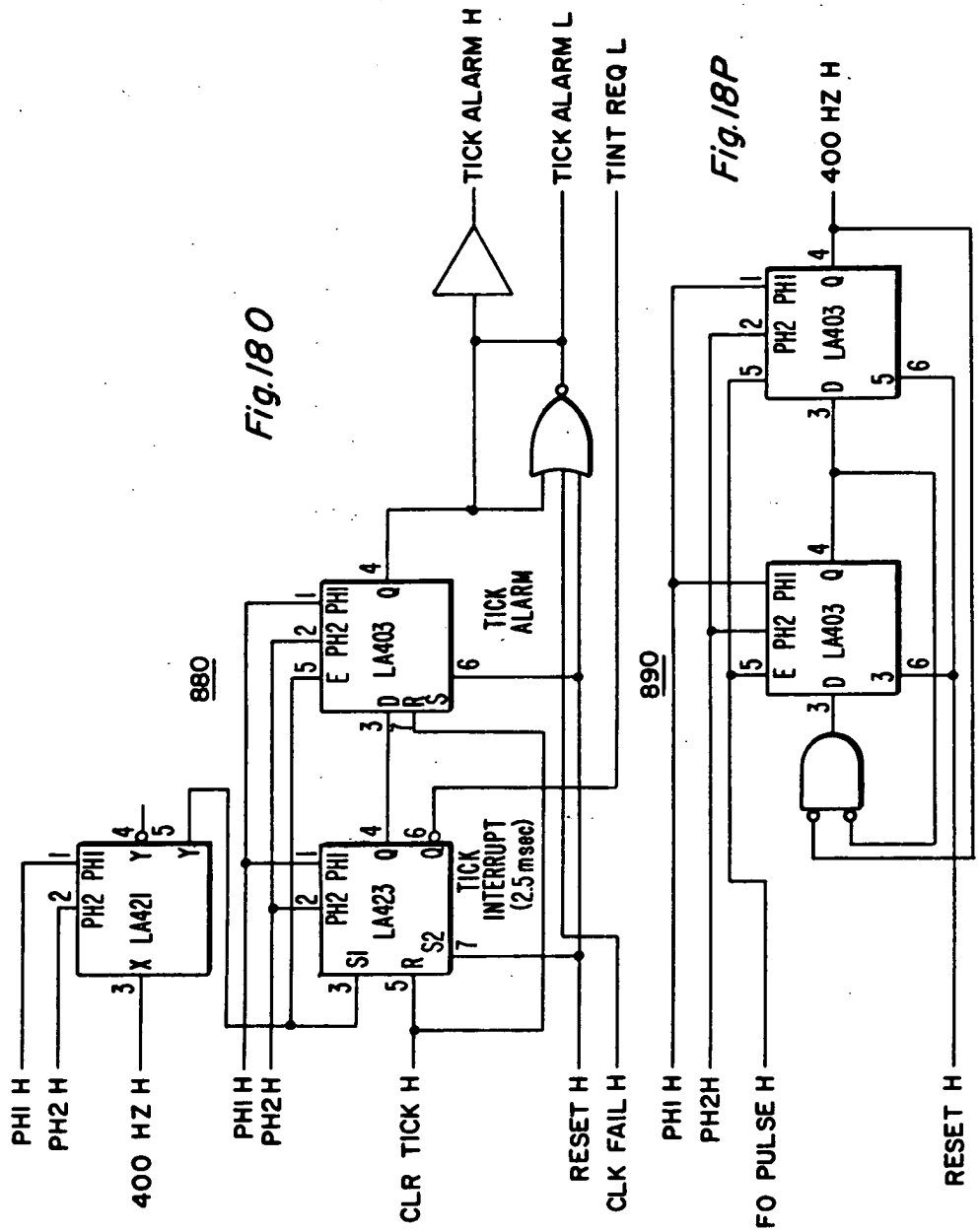


Fig.18O
Fig.18P
880
890
PHI H
PH2 H
400 HZ H
CLR TICK H
RESET H
CLK FAIL H
TICK ALARM H
TICK ALARM L
TINT REQ L
TICK ALARM
TICK INTERRUPT (2.5 msec)
LA421
LA423
LA403
PH1
PH2
PHI
PH2H
FO PULSE H

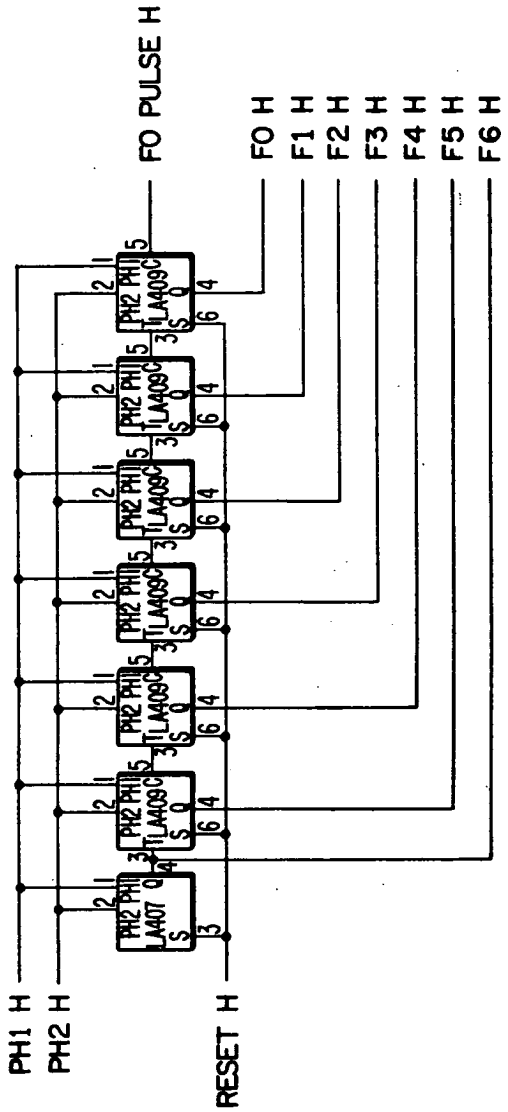


Fig. 18Q

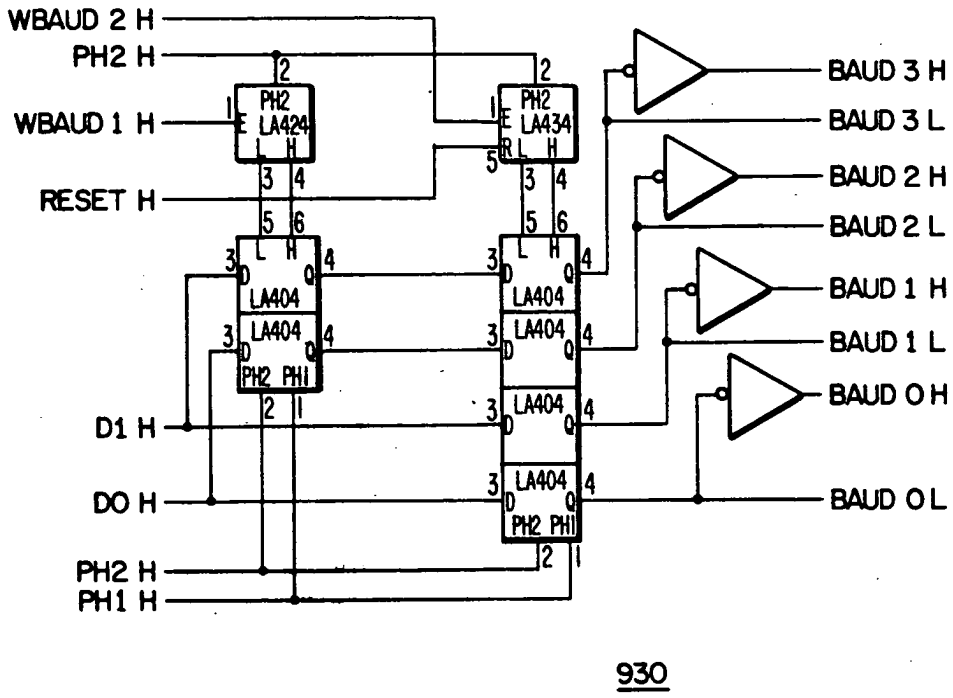

WBAUD 2 H
PH2 H
WBAUD 1 H
RESET H
D1 H
DO H
PH2 H
PH1 H
PH2 LA424
PH2 LA434
LA404
PH2 PH1
BAUD 3 H
BAUD 3 L
BAUD 2 H
BAUD 2 L
BAUD 1 H
BAUD 1 L
BAUD 0 H
BAUD 0 L
930

Fig. 18R

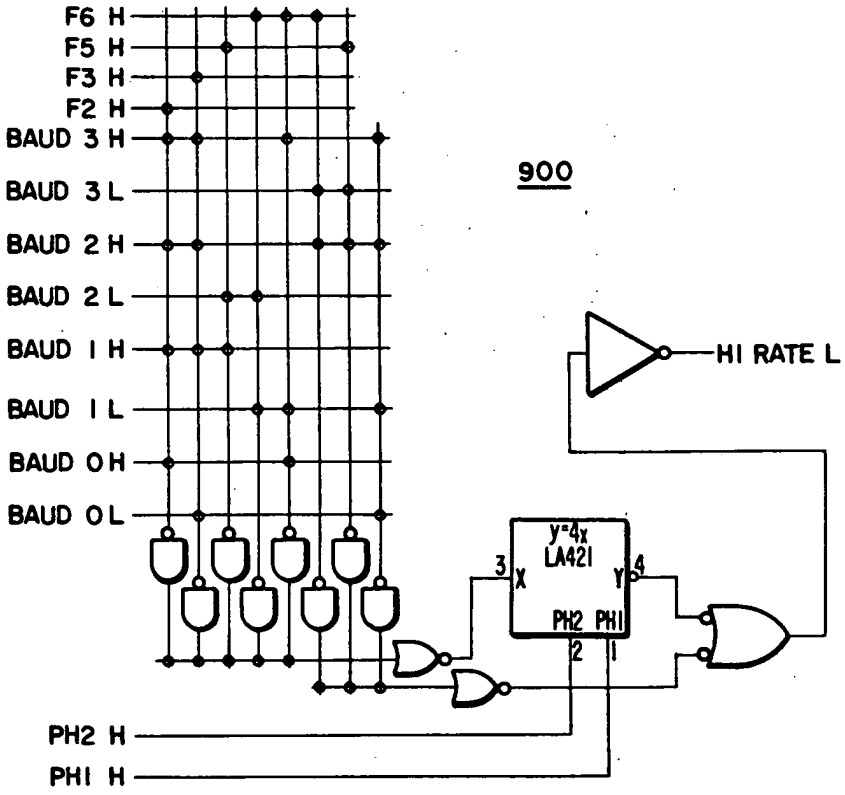

F6 H
F5 H
F3 H
F2 H
BAUD 3 H
BAUD 3 L
BAUD 2 H
BAUD 2 L
BAUD 1 H
BAUD 1 L
BAUD 0 H
BAUD 0 L
900
HI RATE L
y=4x
LA421
3
X
Y
4
PH2 PH1
2
1
PH2 H
PH1 H

Fig. 18S

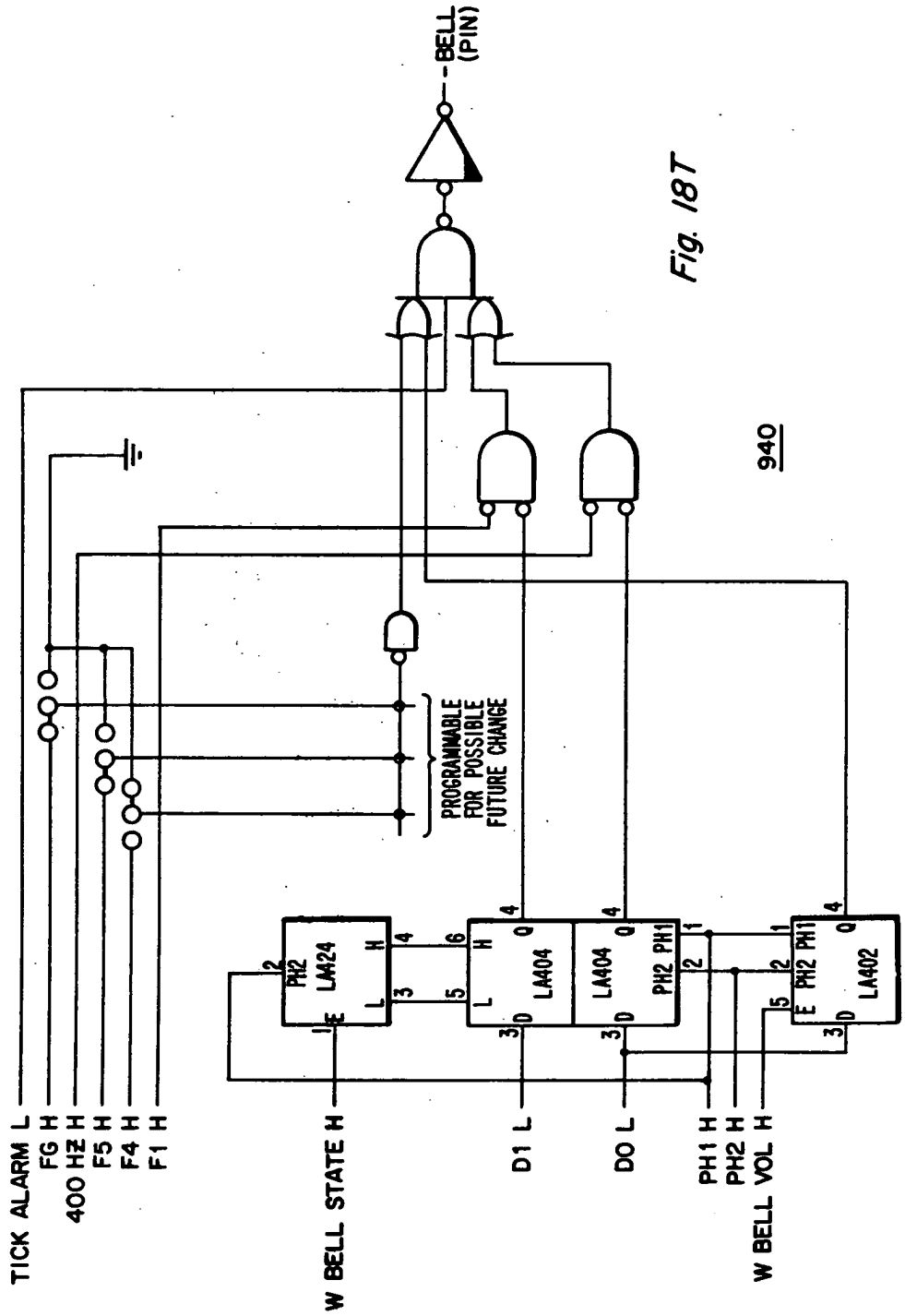

BELL (PIN)
940
PROGRAMMABLE FOR POSSIBLE FUTURE CHANGE
LA424
LA404
LA404
LA402
PH2
PH1
TICK ALARM L
FG H
400 HZ H
F5 H
F4 H
F1 H
W BELL STATE H
D1 L
DO L
PH1 H
PH2 H
W BELL VOL H

Fig. 18T

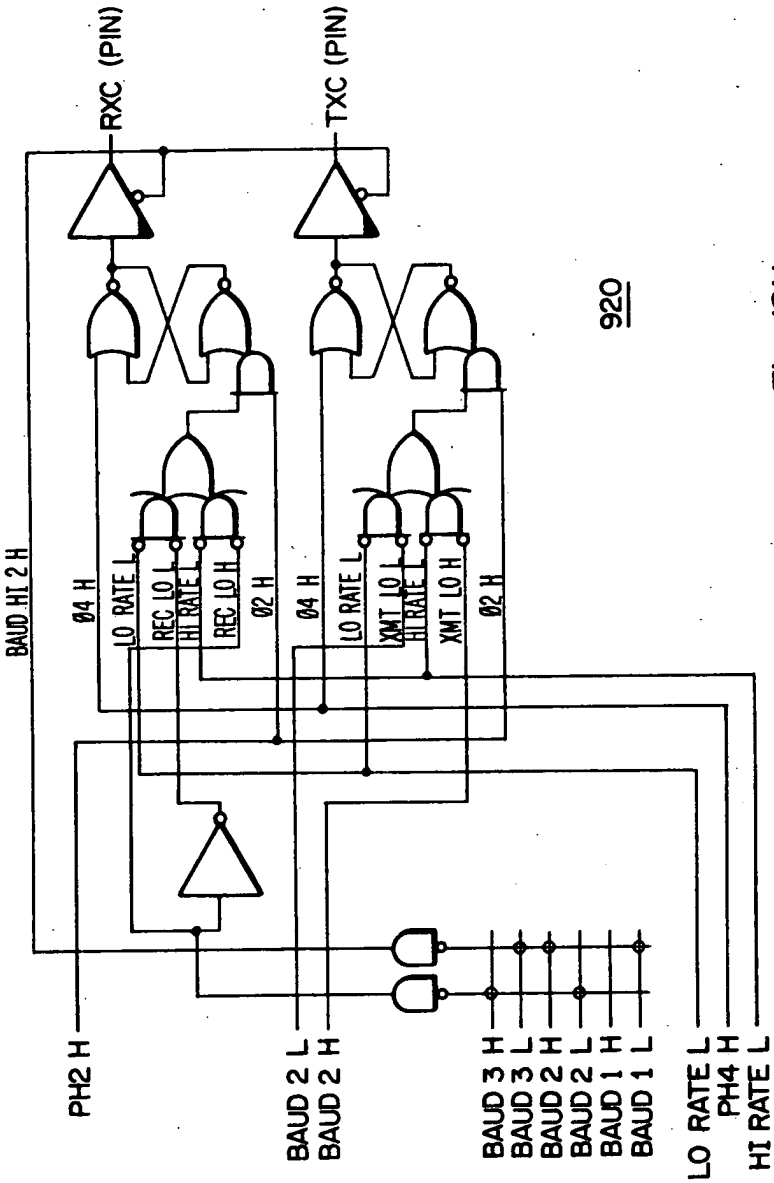

RXC (PIN)
TXC (PIN)
920
BAUD HI 2 H
Ø4 H
LO RATE L
REC LO L
HI RATE L
REC LO H
Ø2 H
Ø4 H
LO RATE L
XMT LO L
HI RATE L
XMT LO H
Ø2 H
PH2 H
BAUD 2 L
BAUD 2 H
BAUD 3 H
BAUD 3 L
BAUD 2 H
BAUD 2 L
BAUD 1 H
BAUD 1 L
LO RATE L
PH4 H
HI RATE L

Fig. 18U

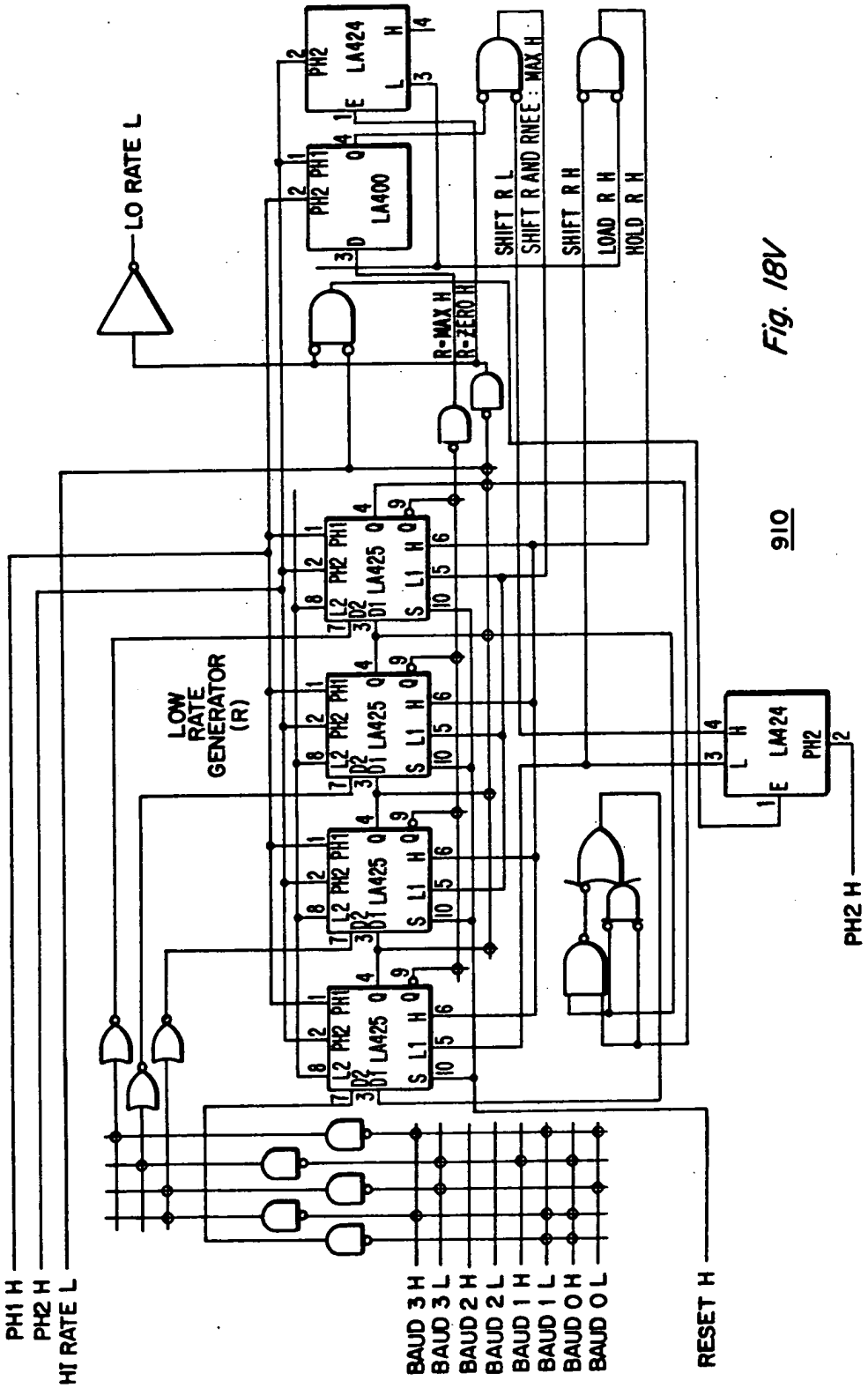

LO RATE L
LA424
LA400
SHIFT R L
SHIFT R AND RNEE: MAX H
SHIFT R H
LOAD R H
HOLD R H
R-MAX H
R-ZERO H
LOW RATE GENERATOR (R)
LA425
LM24
910
PH1 H
PH2 H
HI RATE L
BAUD 3 H
BAUD 3 L
BAUD 2 H
BAUD 2 L
BAUD 1 H
BAUD 1 L
BAUD 0 H
BAUD 0 L
RESET H

Fig. 18V

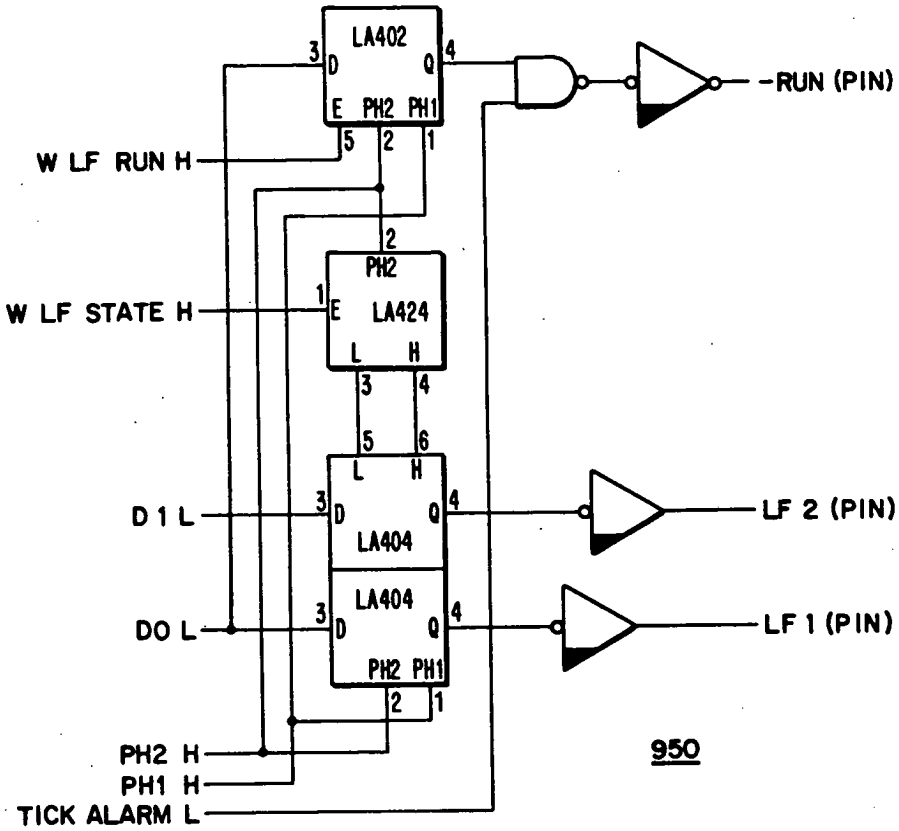

LA402
D
Q
E PH2 PH1
-RUN (PIN)
W LF RUN H
PH2
W LF STATE H
E LA424
L H
L H
D 1 L
D
Q
LA404
LF 2 (PIN)
DO L
LA404
D
Q
PH2 PH1
LF 1 (PIN)
PH2 H
PH1 H
TICK ALARM L
950

Fig. 18W

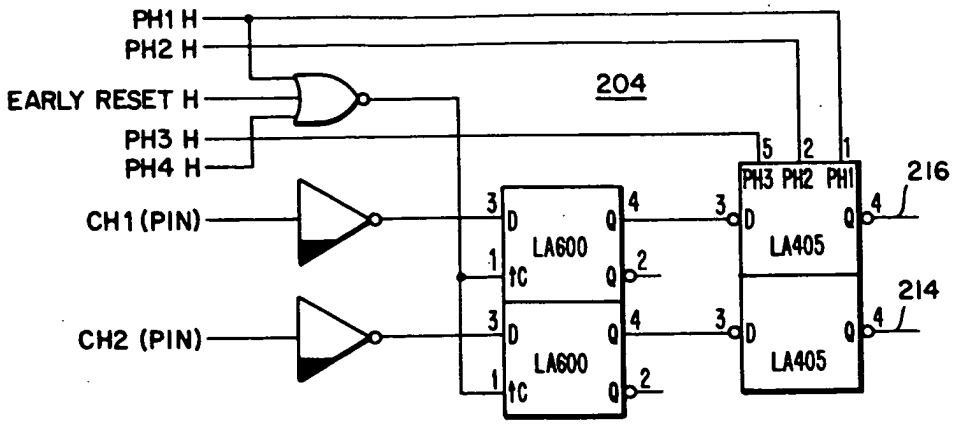


Fig. 18X

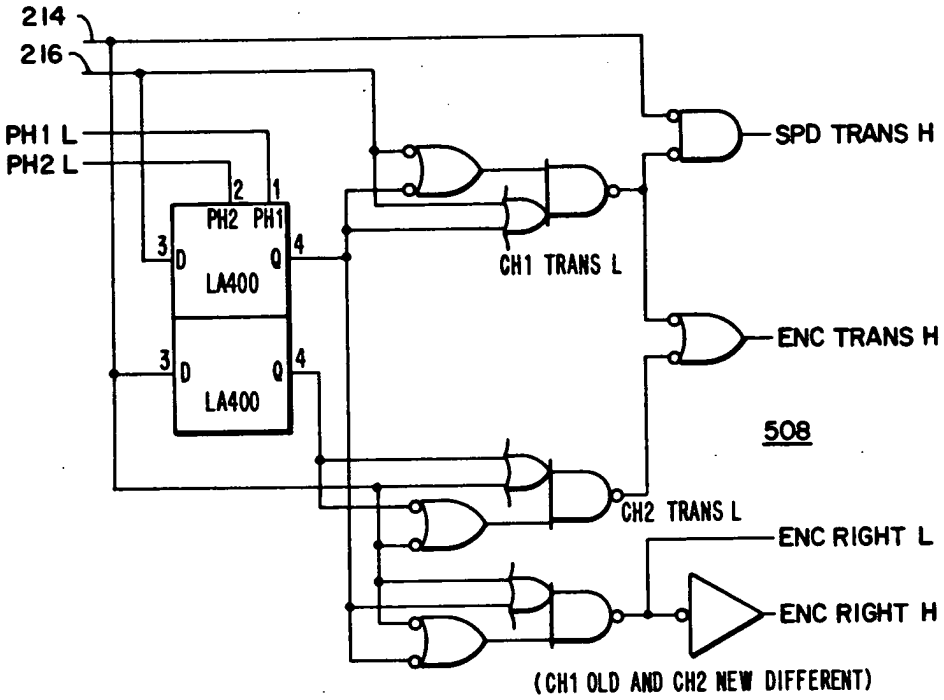


Fig. 18Y

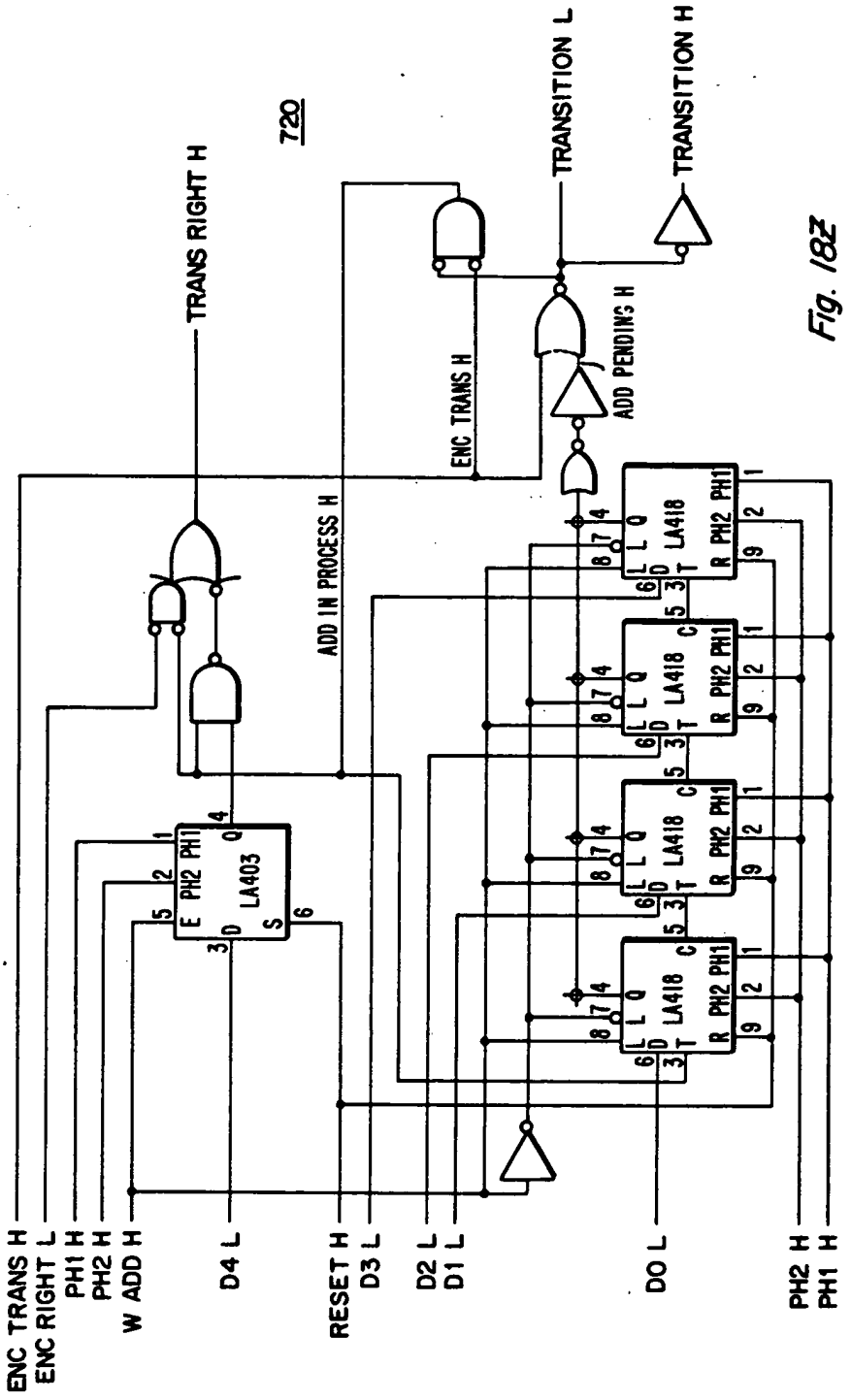

720
TRANS RIGHT H
ADD IN PROCESS H
ENC TRANS H
ADD PENDING H
TRANSITION L
TRANSITION H
LA403
LA418
ENC TRANS H
ENC RIGHT L
PH1 H
PH2 H
W ADD H
D4 L
RESET H
D3 L
D2 L
D1 L
DO L
PH2 H
PH1 H

Fig. 182

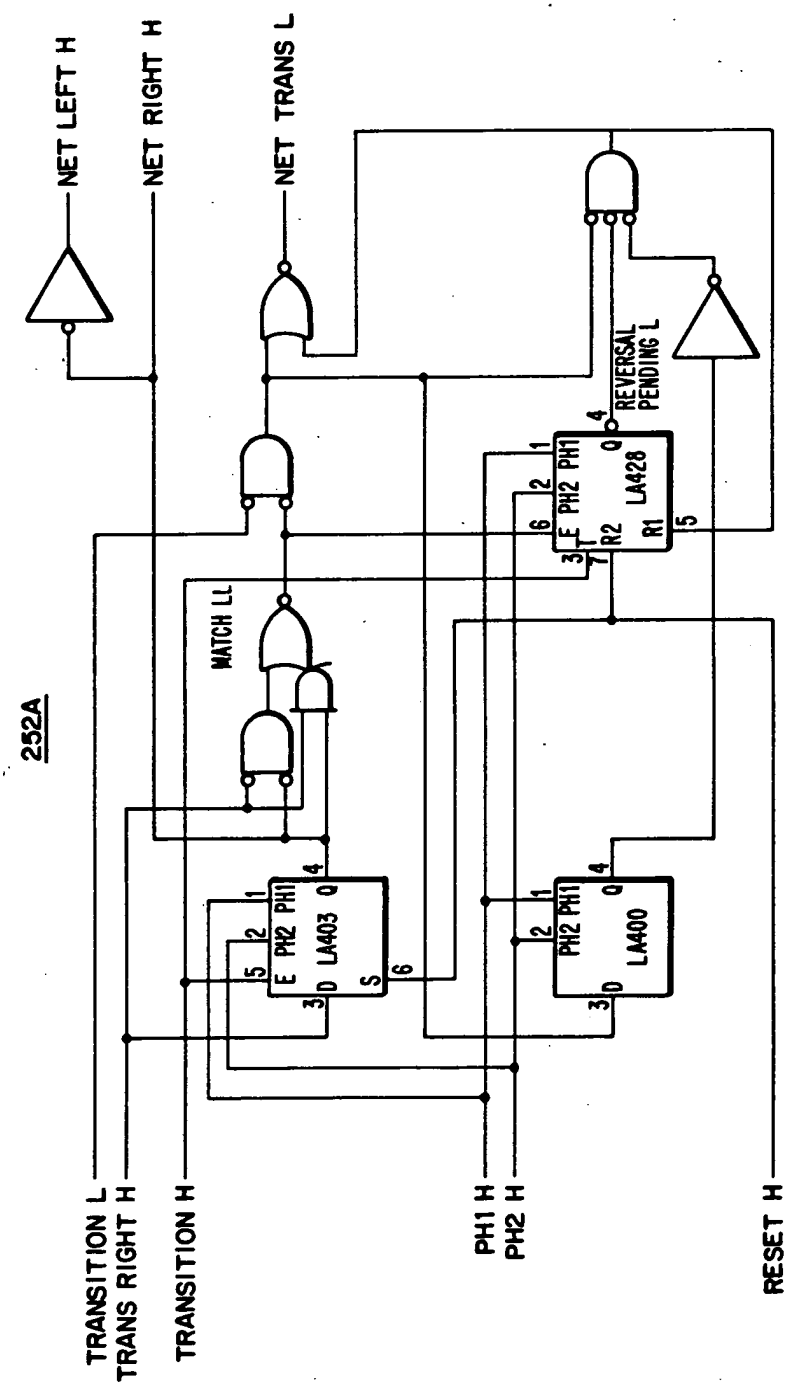
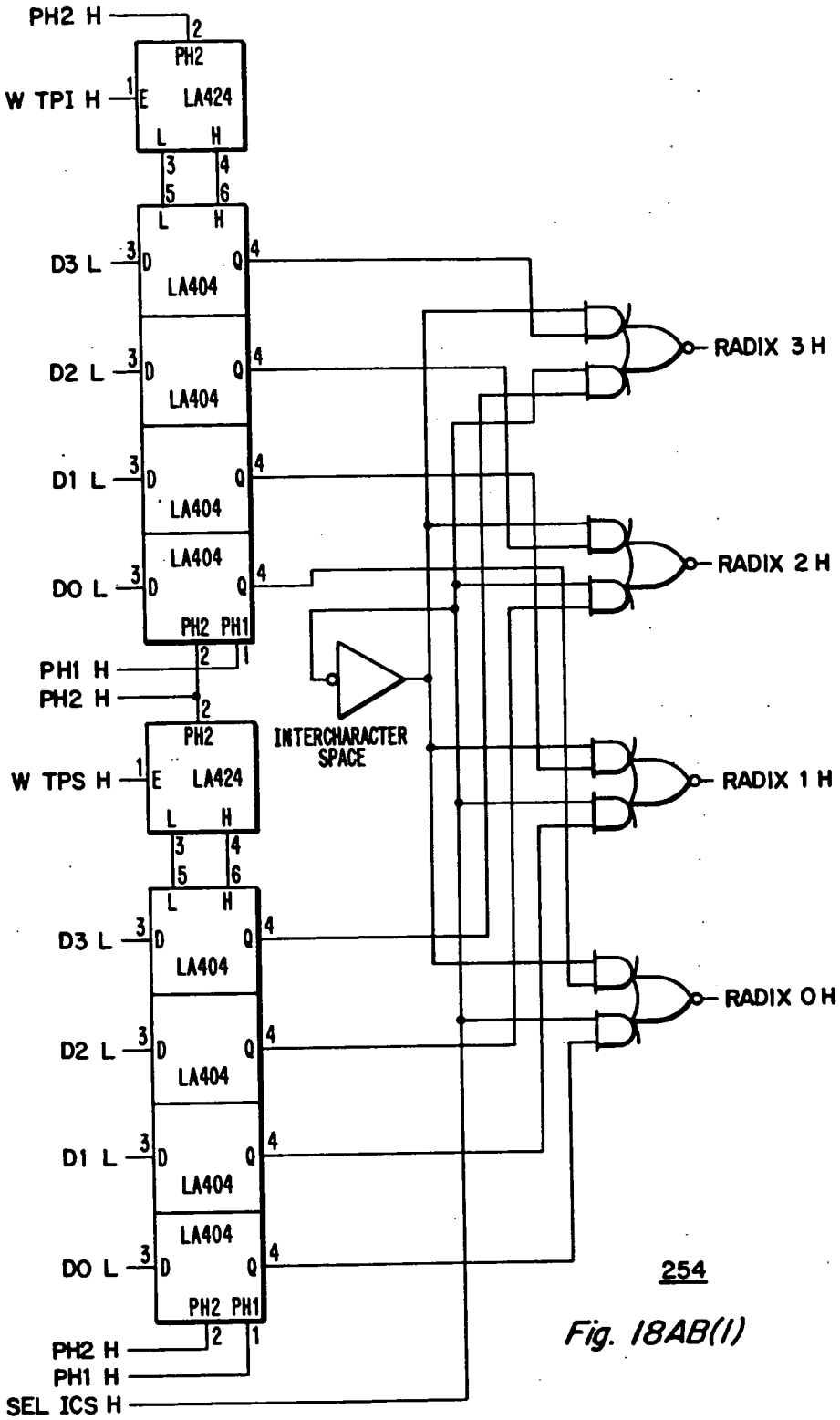

NET LEFT H
NET RIGHT H
NET TRANS L
252A
MATCH LL
REVERSAL PENDING L
LA428
LA403
LA400
PH1
PH2
TRANSITION L
TRANS RIGHT H
TRANSITION H
PH1 H
PH2 H
RESET H

Fig. 18A



254

Fig. 18AB(1)

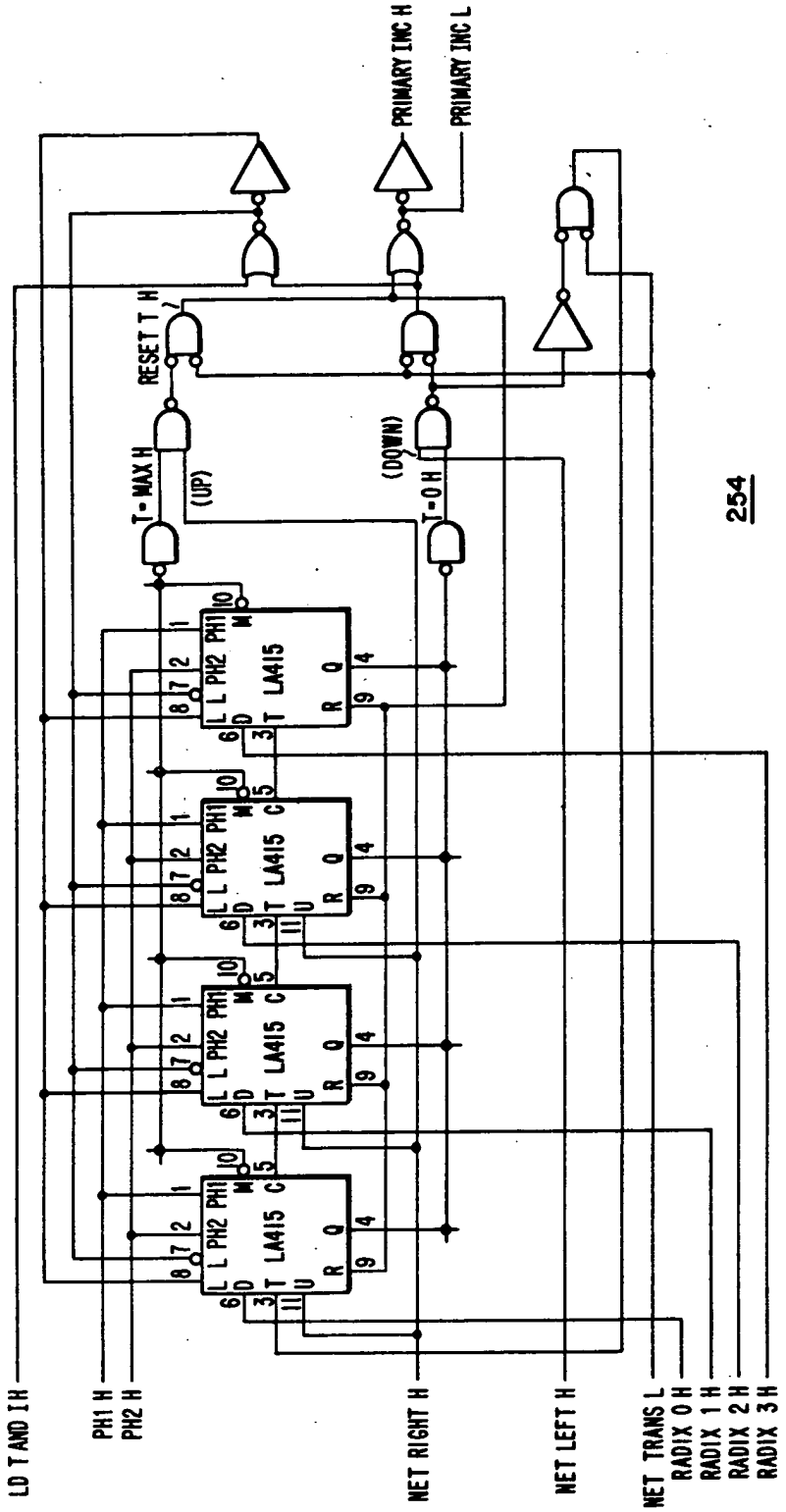

PRIMARY INC H
PRIMARY INC L
RESET T H
T-MAX H
(UP)
(DOWN)
T-0 H
LA415
LD T AND I H
PH1 H
PH2 H
NET RIGHT H
NET LEFT H
NET TRANS L
RADIX 0 H
RADIX 1 H
RADIX 2 H
RADIX 3 H
254

Fig. 18AB(2)

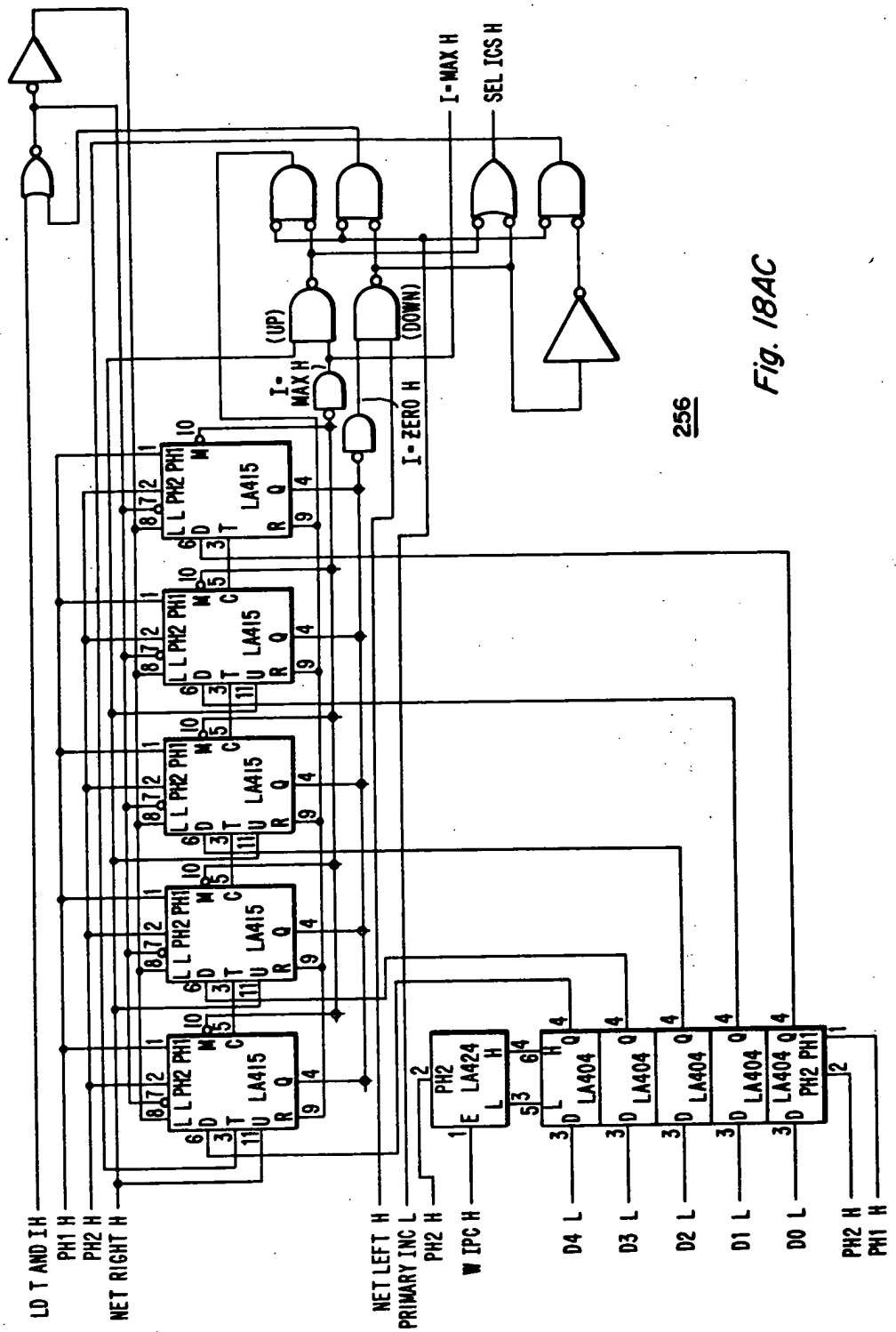

I-MAX H
SEL ICS H
(UP)
(DOWN)
I-MAX H
I-ZERO H
256
LA415
LA424
LA404
PH2
PH1
LD T AND IH
PH1 H
PH2 H
NET RIGHT H
NET LEFT H
PRIMARY INC L
PH2 H
W TPC H
D4 L
D3 L
D2 L
D1 L
D0 L
PH2 H
PH1 H

Fig. 18AC

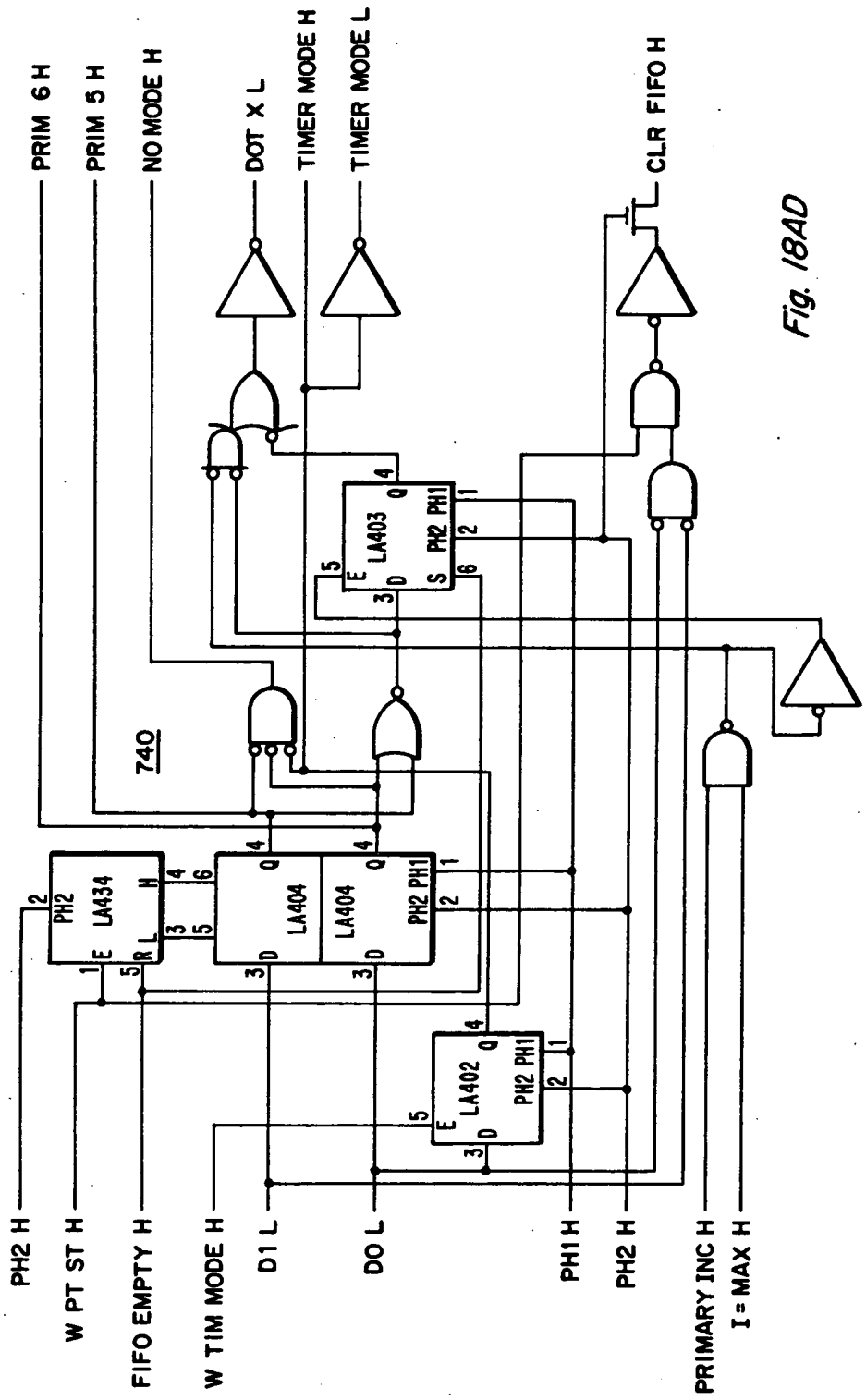

PRIM 6 H
PRIM 5 H
NO MODE H
DOT X L
TIMER MODE H
TIMER MODE L
CLR FIFO H
740
LA434
LA404
LA404
LA403
LA402
PH2
PH1
PH2 H
W PT ST H
FIFO EMPTY H
W TIM MODE H
D1 L
DO L
PH1 H
PH2 H
PRIMARY INC H
I = MAX H

Fig. 184D

Fig. 18AF

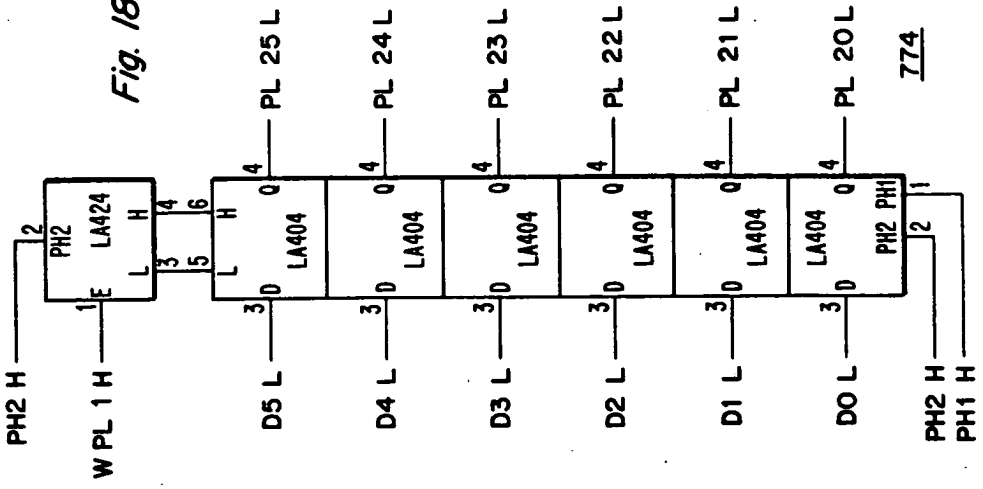
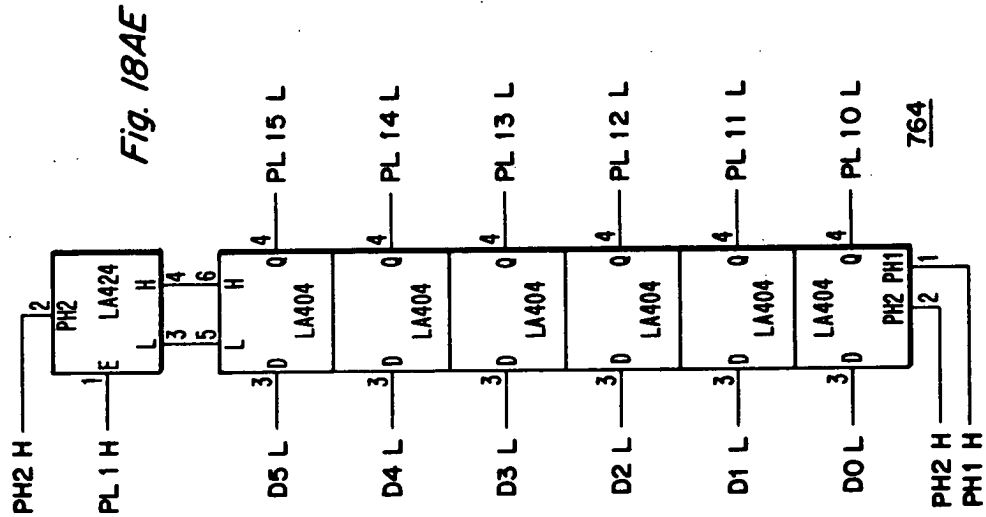


Fig. 18AE



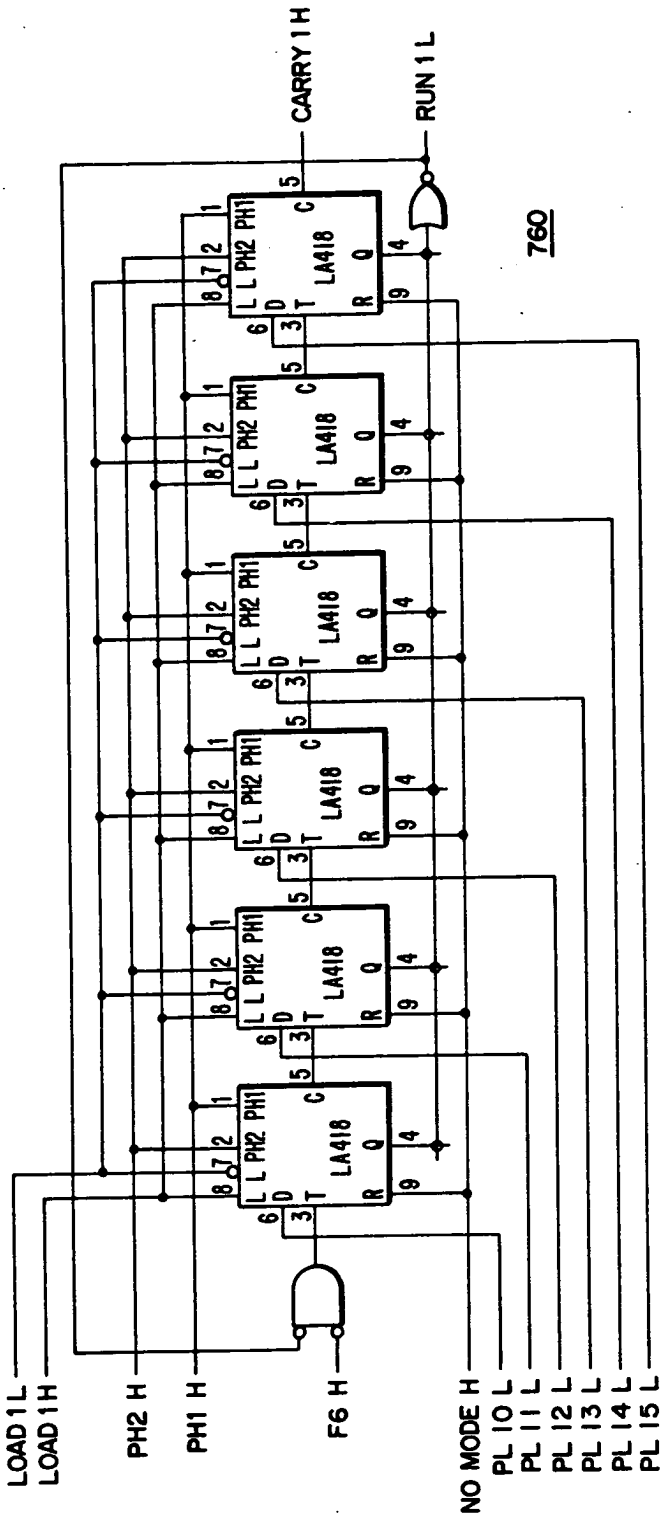

CARRY 1 H
RUN 1 L
760
LA418
LOAD 1 L
LOAD 1 H
PH2 H
PH1 H
F6 H
NO MODE H
PL 10 L
PL 11 L
PL 12 L
PL 13 L
PL 14 L
PL 15 L

Fig. 18AG

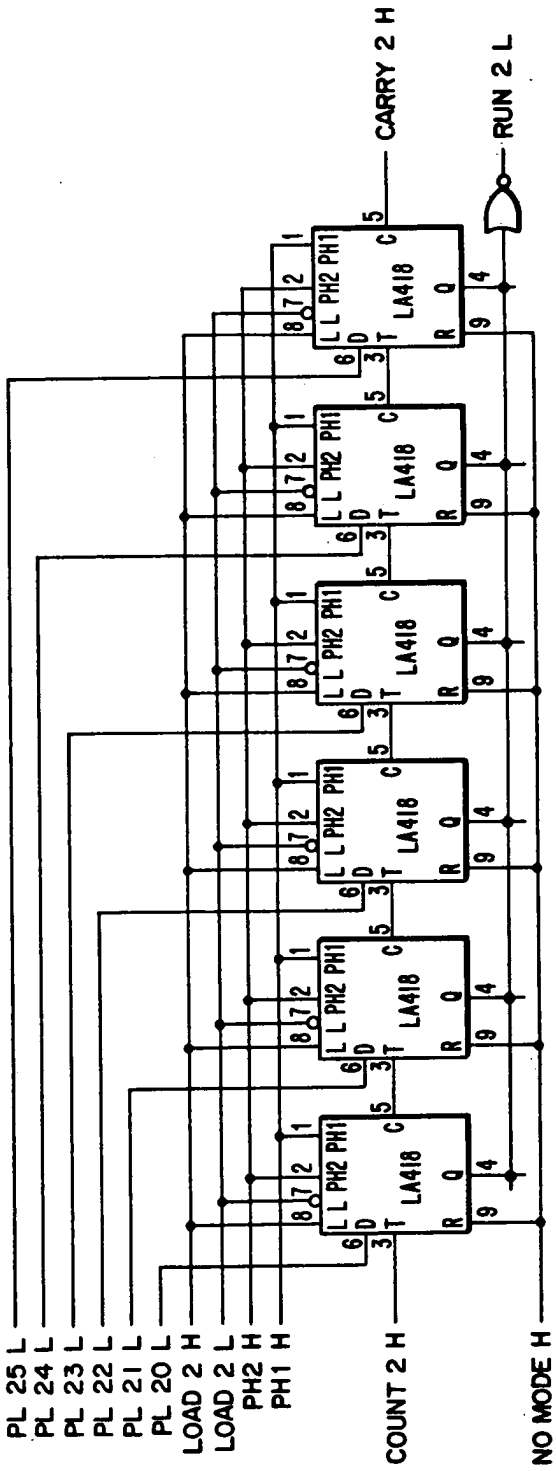

PL 25 L
PL 24 L
PL 23 L
PL 22 L
PL 21 L
PL 20 L
LOAD 2 H
LOAD 2 L
PH2 H
PH1 H
COUNT 2 H
NO MODE H
CARRY 2 H
RUN 2 L
LA418
L L PH2 PH1
D
T
R
Q
C
770

Fig. 18AH

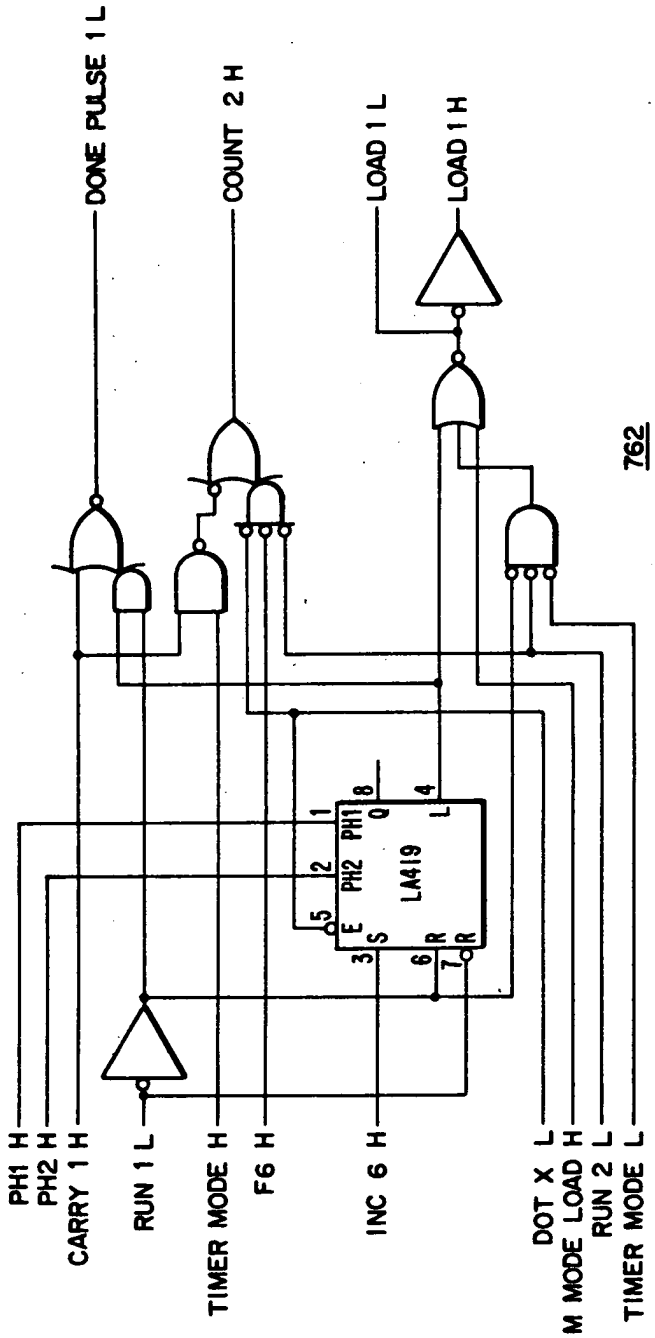

DONE PULSE 1 L
COUNT 2 H
LOAD 1 L
LOAD 1 H
762
LA419
PH1
PH2
PH1 H
PH2 H
CARRY 1 H
RUN 1 L
TIMER MODE H
F6 H
INC 6 H
DOT X L
TIM MODE LOAD H
RUN 2 L
TIMER MODE L

Fig. 184I

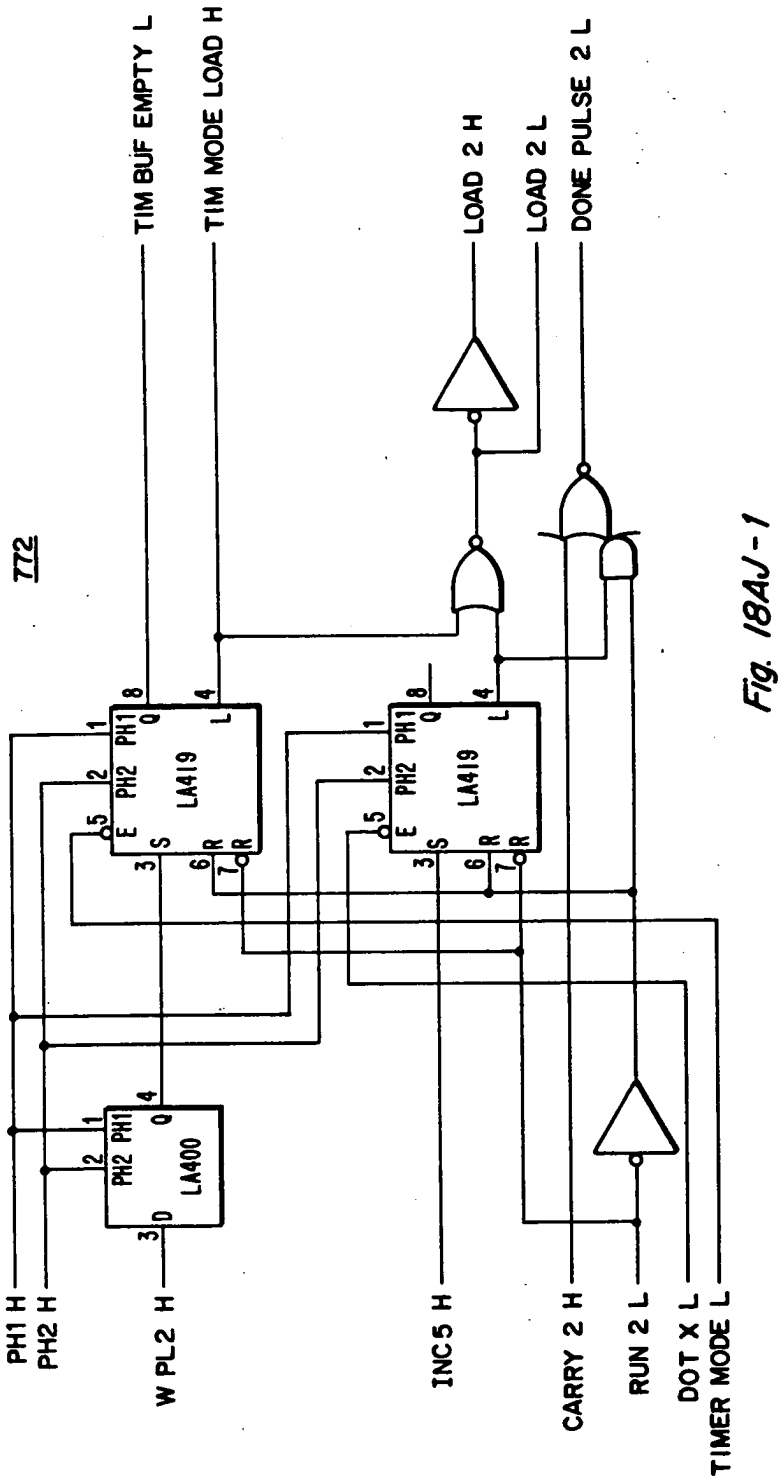

772
TIM BUF EMPTY L
TIM MODE LOAD H
LOAD 2 H
LOAD 2 L
DONE PULSE 2 L
LA419
LA400
PH1 H
PH2 H
W PL2 H
INC5 H
CARRY 2 H
RUN 2 L
DOT X L
TIMER MODE L

Fig. 184J-1

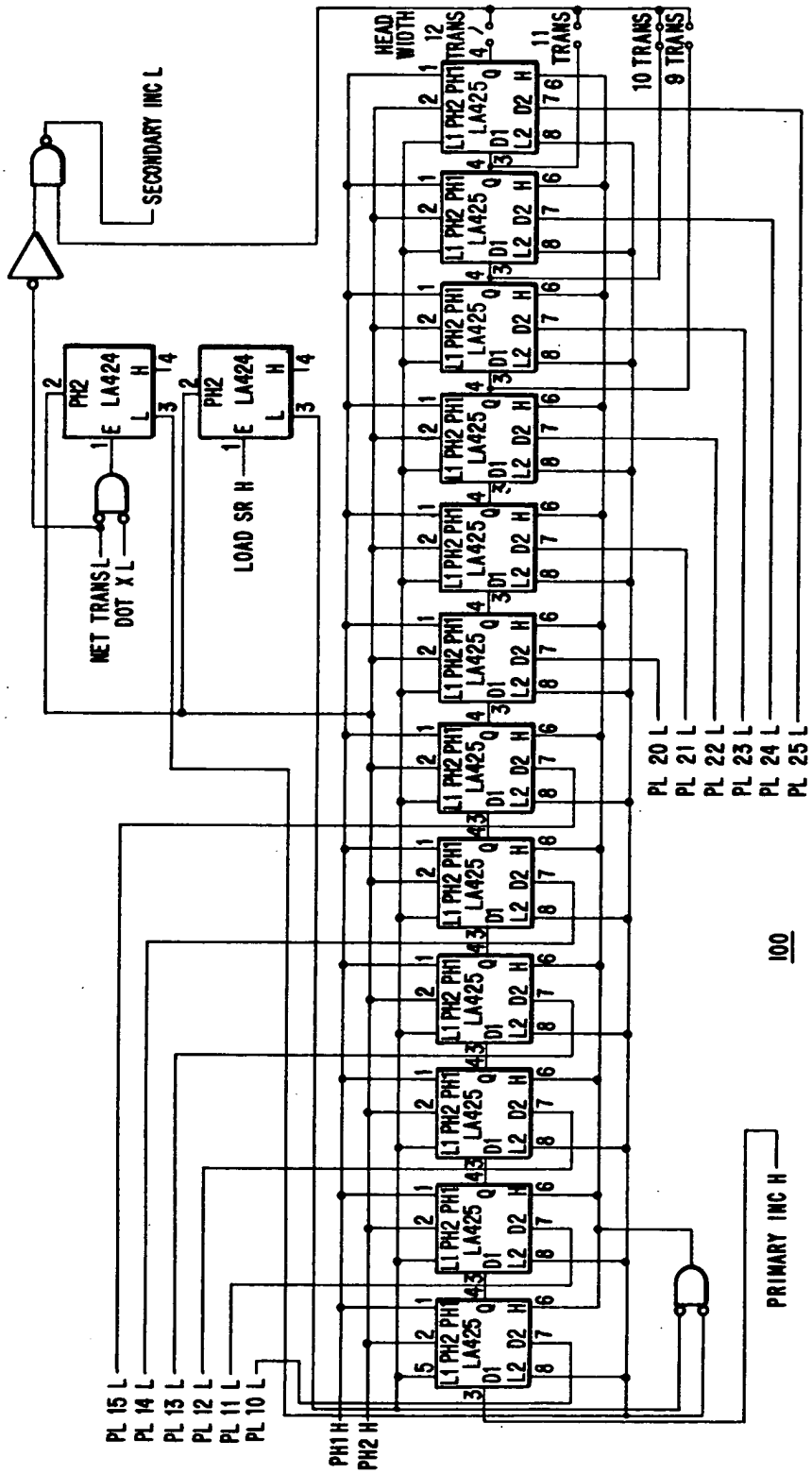

SECONDARY INC L
HEAD WIDTH
12 TRANS
11 TRANS
10 TRANS
9 TRANS
PH2
LA424
LOAD SR H
NET TRANS L
DOT X L
L1 PH2 PH1
LA425
L2 D2 H
PL 20 L
PL 21 L
PL 22 L
PL 23 L
PL 24 L
PL 25 L
PL 15 L
PL 14 L
PL 13 L
PL 12 L
PL 11 L
PL 10 L
PH1 H
PH2 H
PRIMARY INC H
100

Fig. 18AJ-2

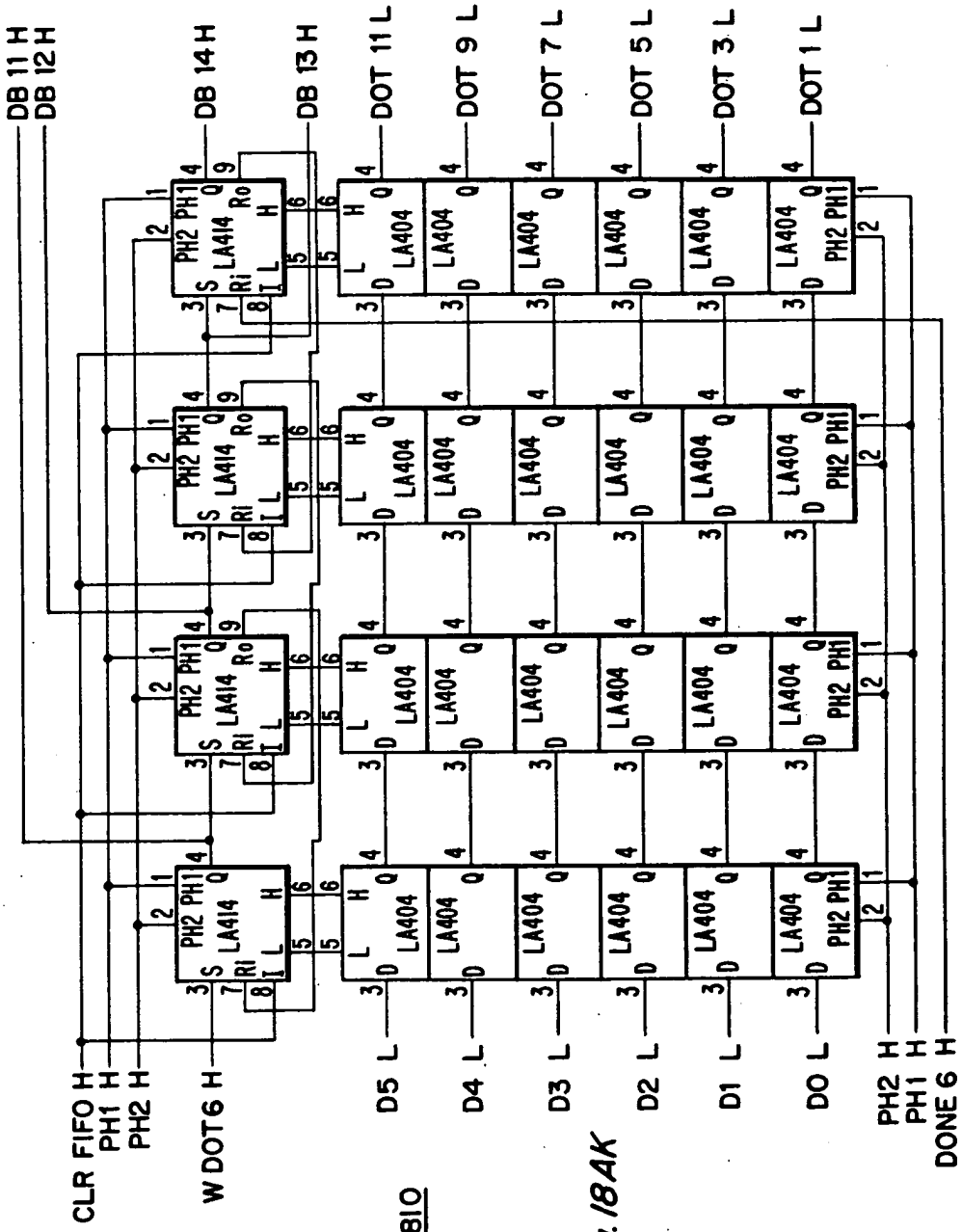

DB 11 H
DB 12 H
DB 14 H
DB 13 H
DOT 11 L
DOT 9 L
DOT 7 L
DOT 5 L
DOT 3 L
DOT 1 L
LA414
LA404
PH2 PH1
CLR FIFO H
PH1 H
PH2 H
W DOT6 H
D5 L
D4 L
D3 L
D2 L
D1 L
DO L
PH2 H
PH1 H
DONE 6 H
810

Fig. 18AK

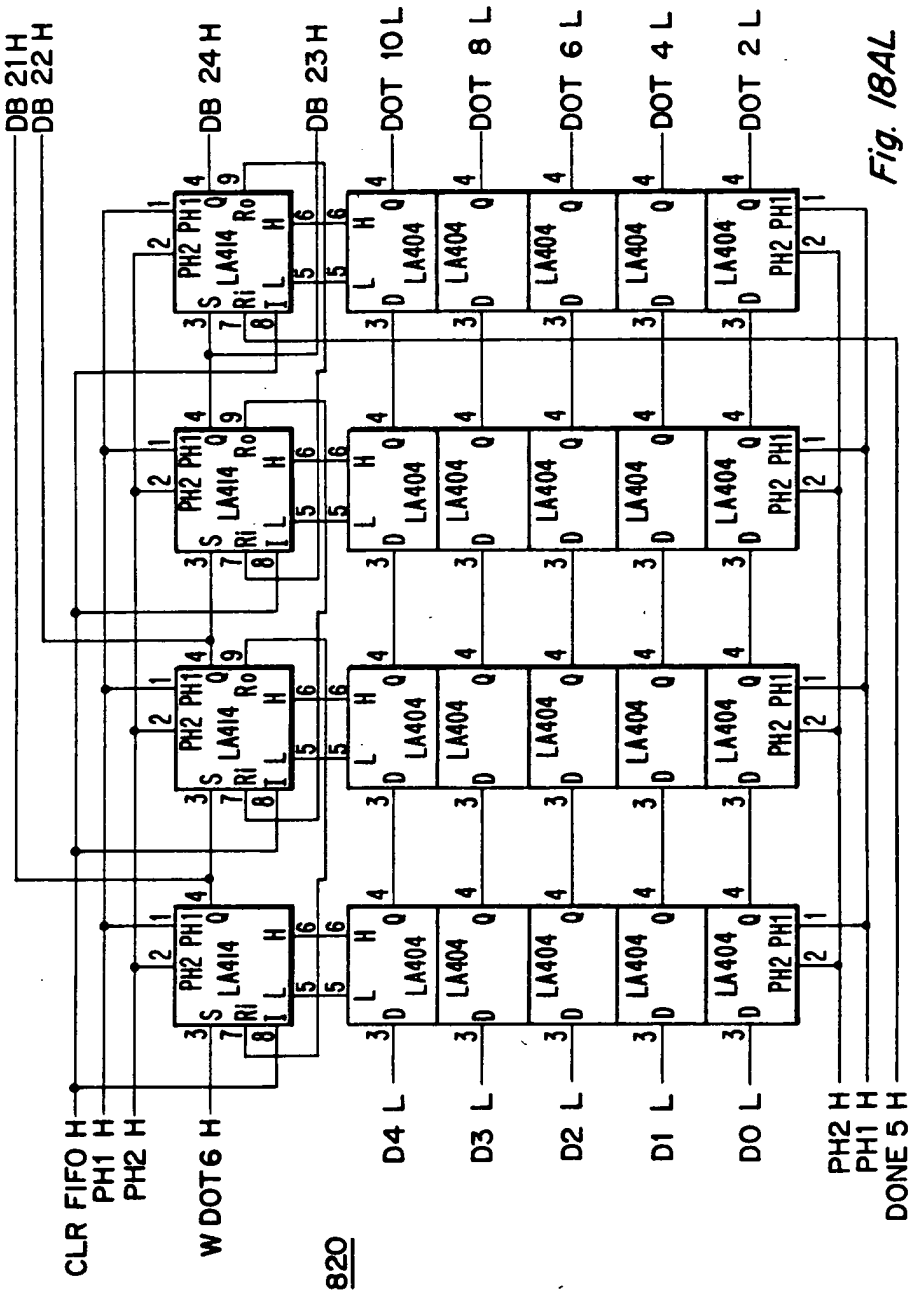

DB 21H
DB 22H
DB 24H
DB 23H
DOT 10L
DOT 8L
DOT 6L
DOT 4L
DOT 2L
LA414
LA404
PH2 PH1
CLR FIFO H
PH1 H
PH2 H
W DOT 6 H
D4 L
D3 L
D2 L
D1 L
DO L
PH2 H
PH1 H
DONE 5 H
820

Fig. 184L

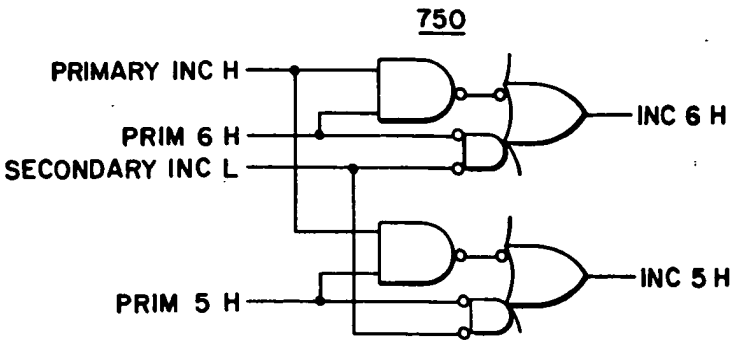


Fig. 18 AM

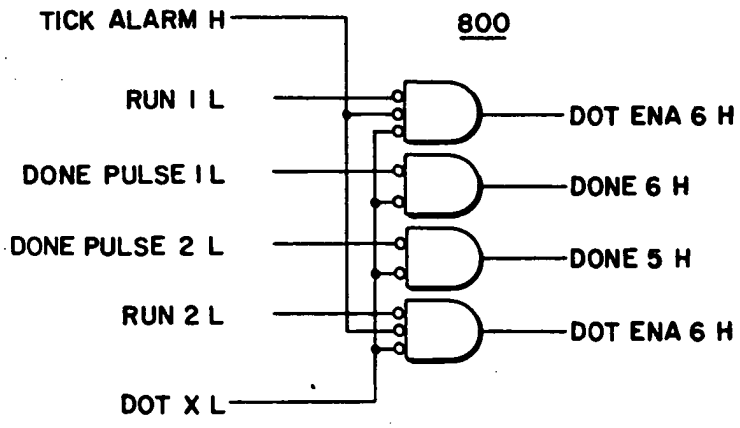


Fig. 18 AN

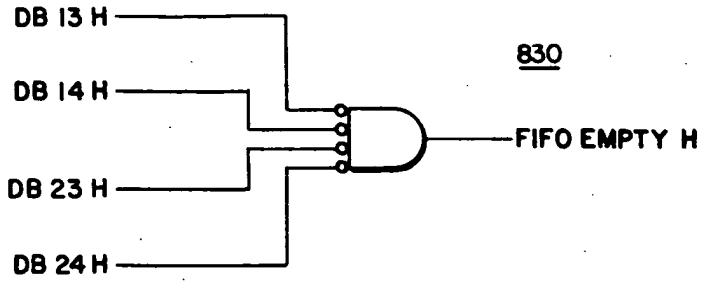


Fig. 18 AO

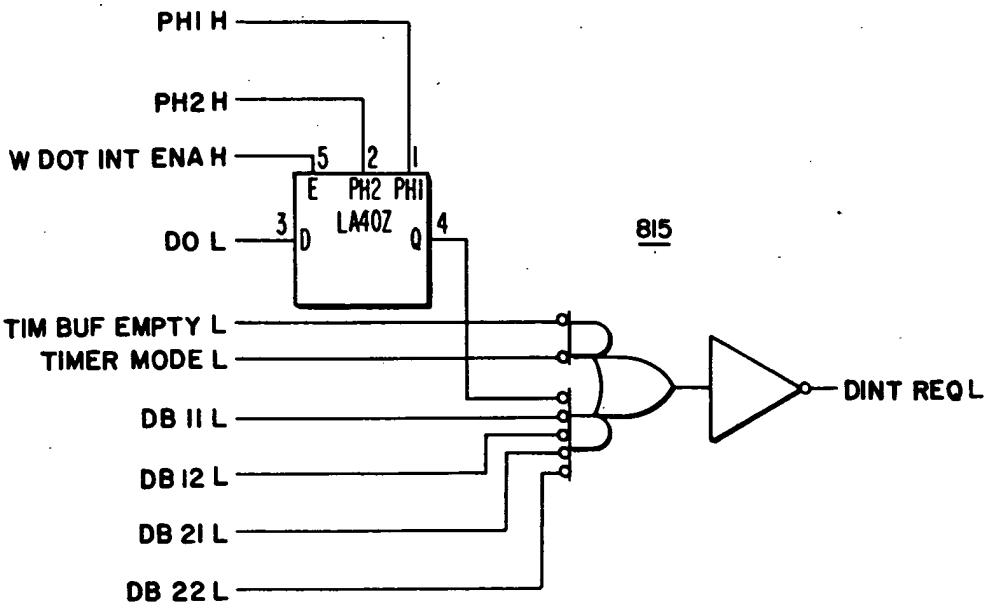


Fig. 18 AP

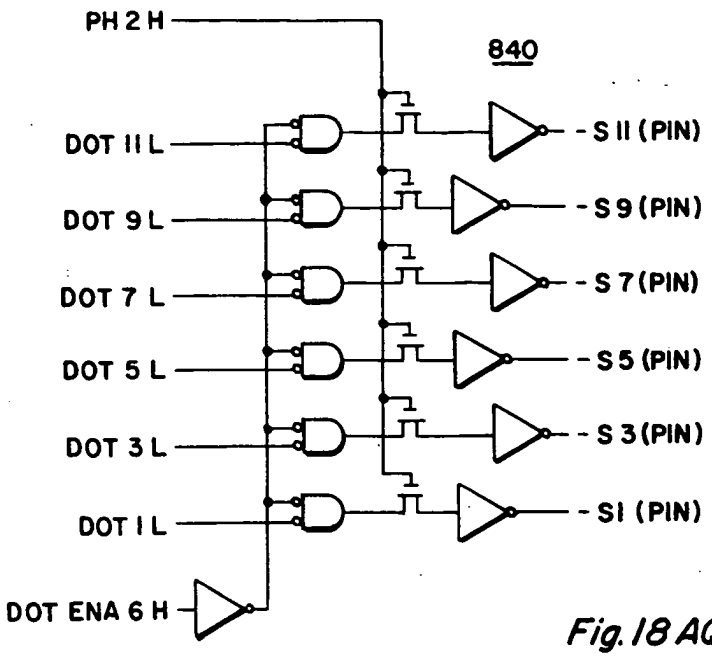


Fig. 18 AQ

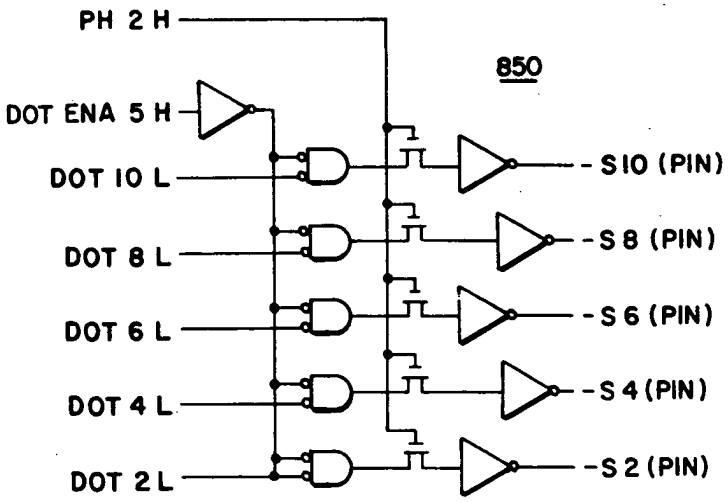


Fig. 18 AR

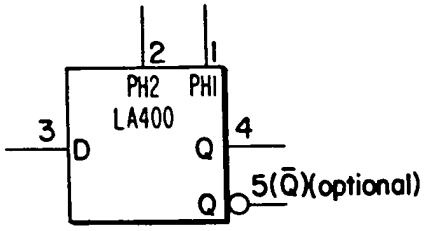


Fig. 19A(1)

Dn	Qn+1
0	0
1	1

Fig. 19A(2)

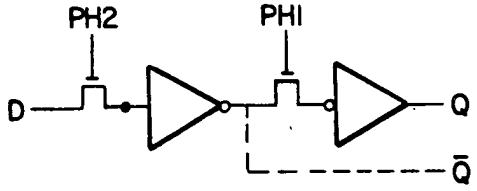


Fig. 19A(3)

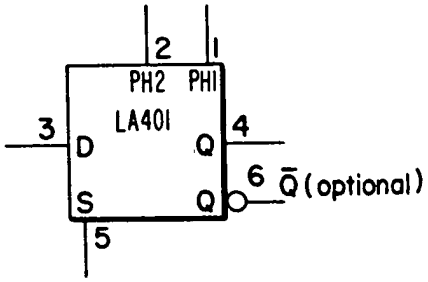


Fig. 19B(1)

Sn	Dn	Qn+1
0	0	0
X	1	1
1	X	1

Fig. 19B(2)

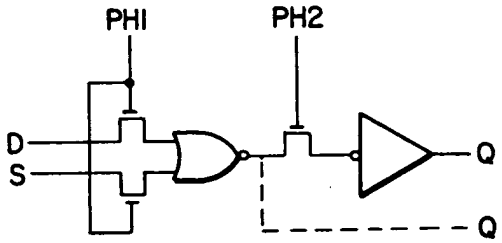


Fig. 19B(3)

Fig. 19C(1)

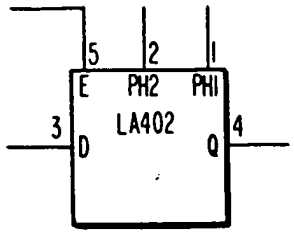


Fig. 19C(2)

En	Dn	Qn	Qn+1
0	X	0	0
0	X	1	1
1	0	X	0
1	1	X	1

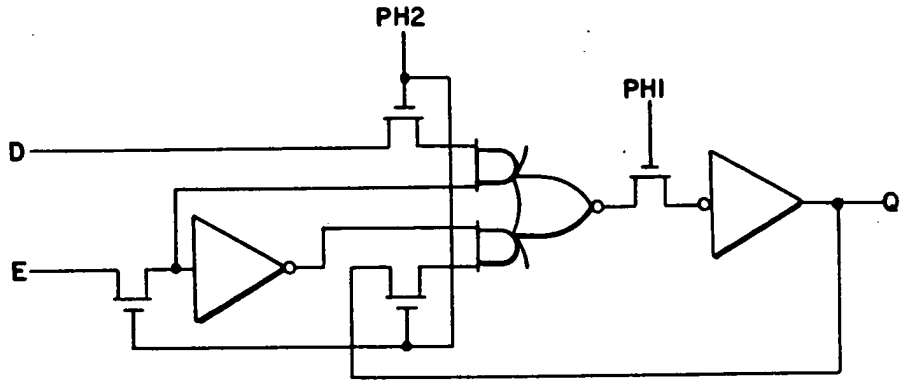


Fig. 19C(3)

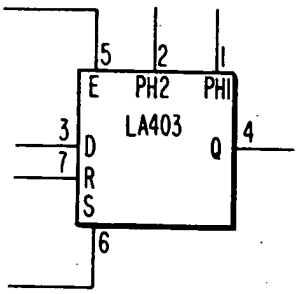


Fig. 19D(1)

Rn	Sn	En	Dn	Qn	Qn+1
0	0	0	X	0	0
0	0	0	X	1	1
X	0	1	0	X	0
X	0	1	1	X	1
X	1	X	X	X	1
1	0	0	X	X	0

Fig. 19D(2)

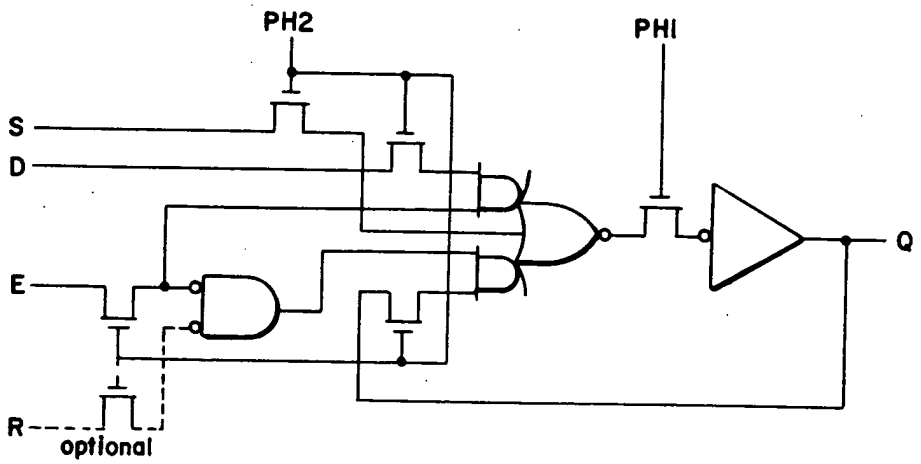


Fig. 19D(3)

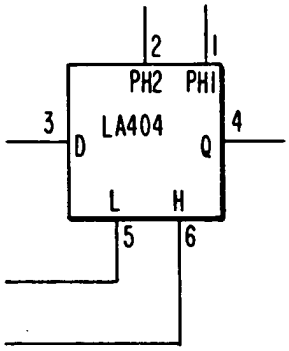


Fig. 19E (1)

Dn	Ln	Hn	Qn	Qn+1
X	0	0	X	0
X	0	1	0	0
0	1	0	X	0
0	1	1	0	0
1	1	X	X	1
X	X	1	1	1

Fig. 19E (2)

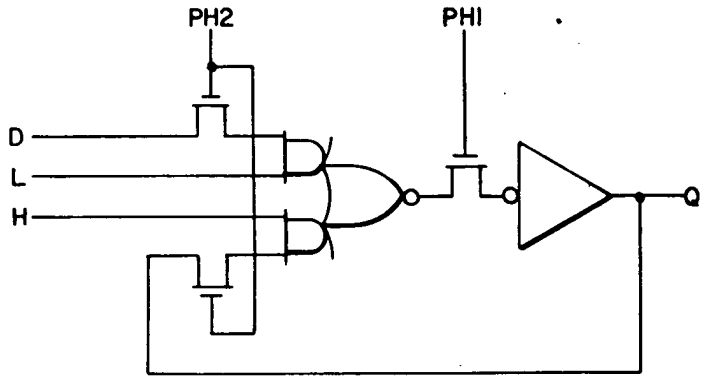


Fig. 19E (3)

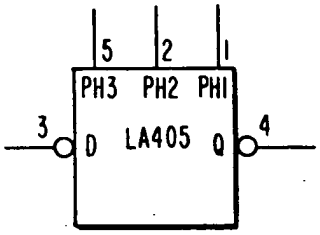


Fig. 19F(1)

Qn	Dn-1/2	Dn	Qn+1
0	0	X	0
0	X	0	0
0	1	1	1
1	1	X	1
1	X	1	1
1	0	0	0

Fig. 19F(2)

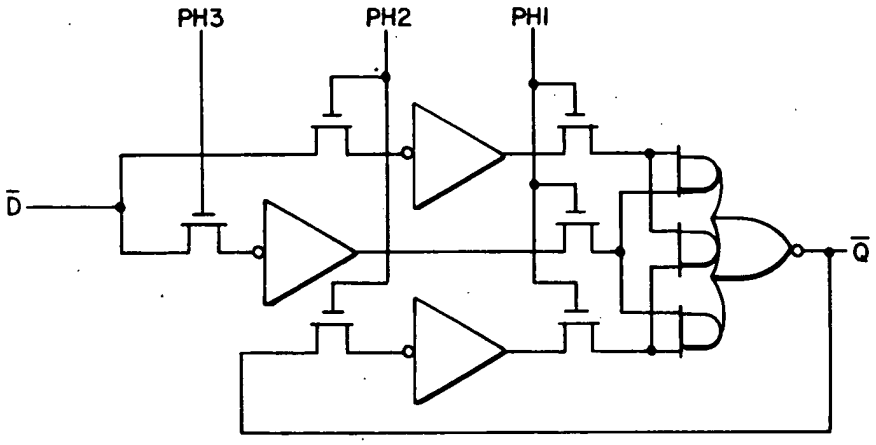


Fig. 19F(3)

Fig. 19G(1)

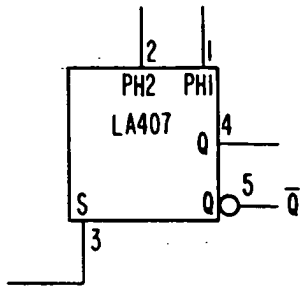


Fig. 19G(2)

Sn	Qn	Qn+1
0	0	1
0	1	0
1	X	1

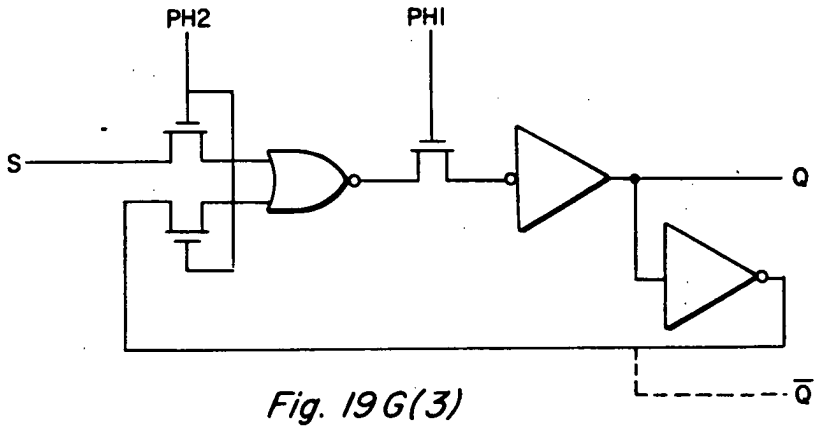


Fig. 19G(3)

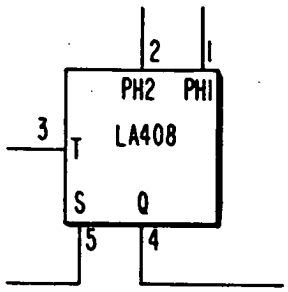


Fig. 19H(1)

Sn	Tn	Qn	Qn+1
0	0	0	0
X	0	1	1
X	1	0	1
0	1	1	0
1	X	X	1

Fig. 19H(2)

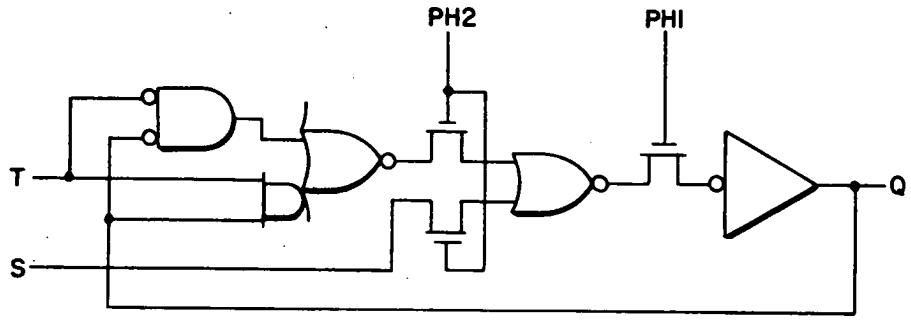


Fig. 19H(3)

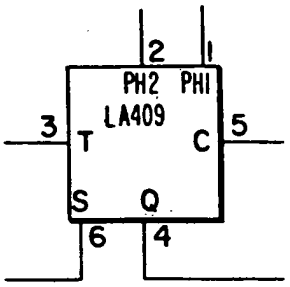


Fig. 191(1)

Sn	Tn	Qn	Cn	Qn+1
0	0	0	0	0
X	0	1	0	1
X	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	1	1	1	1

Fig. 191(2)

Tn ⟶ Cn : 2 LEVELS
Qn ⟶ Cn : 2 LEVELS

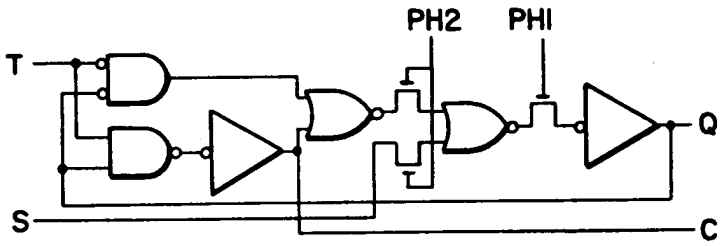


Fig. 191(3)

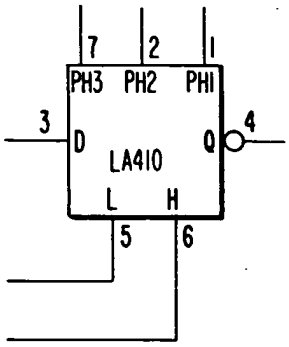


Fig. 19J(1)

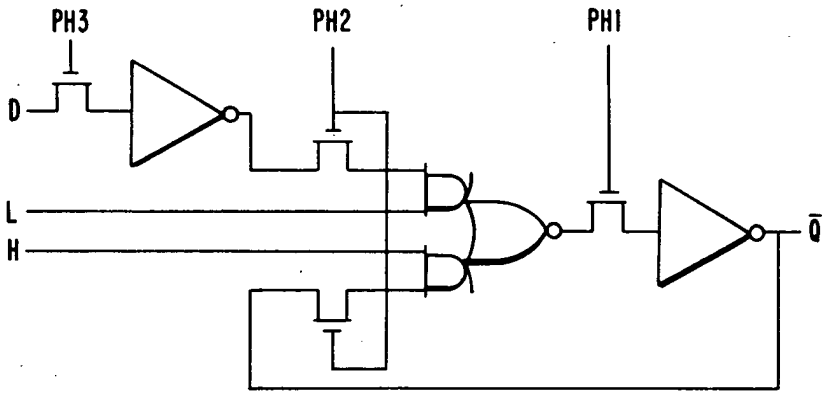


Fig. 19J(2)

Fig. 19K(2)

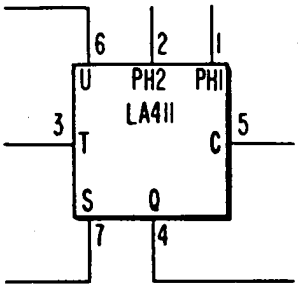


Fig. 19K(1)

Sn	Un	Tn	Qn	Cn	Qn+1
0	X	0	0	0	0
X	X	0	1	0	1
X	0	1	0	1	1
0	0	1	1	0	0
X	1	1	0	0	1
0	1	1	1	1	0
1	0	1	1	0	1
1	1	1	1	1	1

Tn ⟶ Cn: 2 LEVELS

Un ⟶ Cn: 3 LEVELS

Qn ⟶ Cn: 3 LEVELS

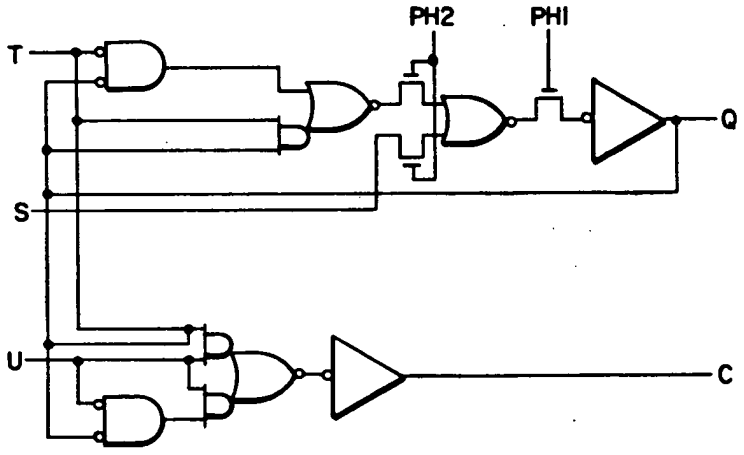


Fig. 19K(3)

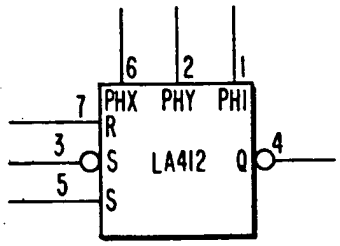


Fig. 19L(1)

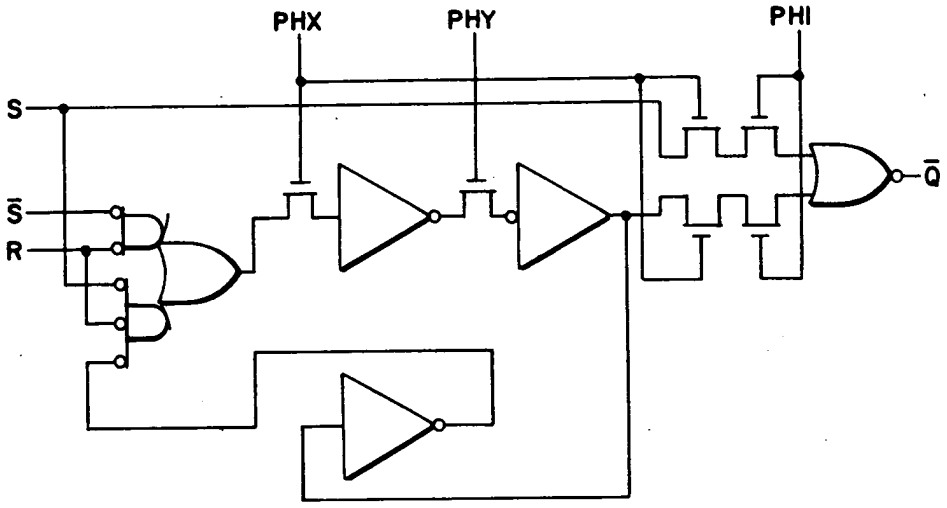


Fig. 19L(2)

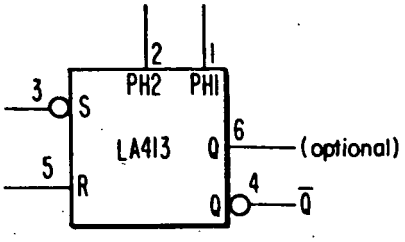


Fig. 19M(1)

S̄	R	Qn	Qn+1
1	0	0	0
X	0	1	1
1	1	X	0
0	X	X	1

Fig. 19M(2)

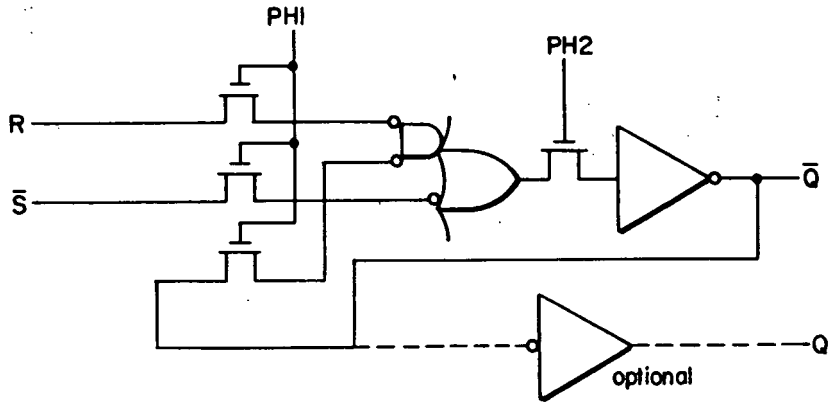


Fig. 19M(3)

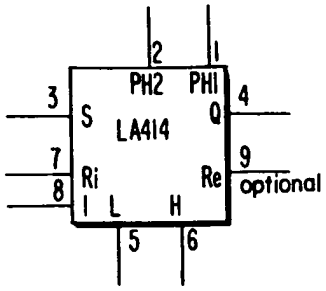


Fig. 19N (1)

In	Rn	Sn	Qn	Qn-1	Ln-1/2	Hn-1/2	Ron
0	X	0	0	0	0	1	0
0	0	0	1	1	0	1	0
0	X	1	0	1	1	0	1
0	0	1	1	1	0	1	0
0	1	0	1	0	0	1	0
0	1	1	1	1	1	0	1
1	X	X	X	0	X	X	X

Fig. 19N (2)

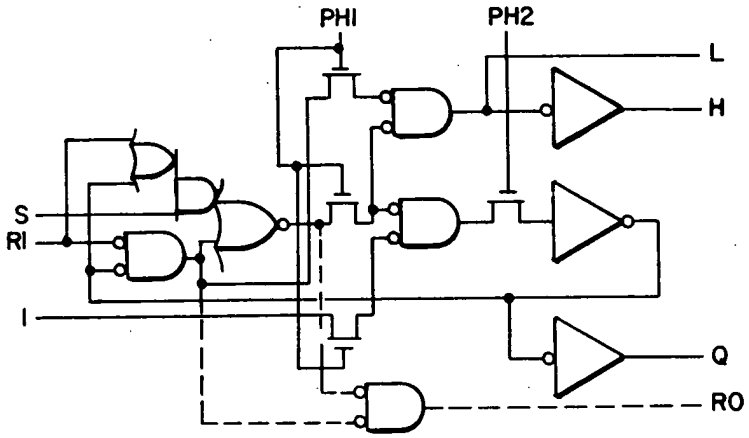


Fig. 19N (3)

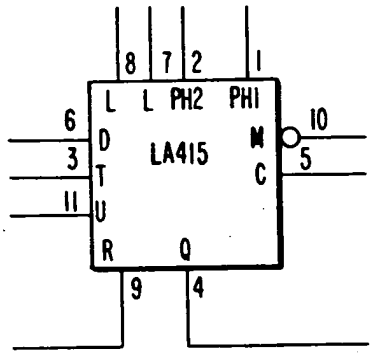


Fig. 19 0 (1)

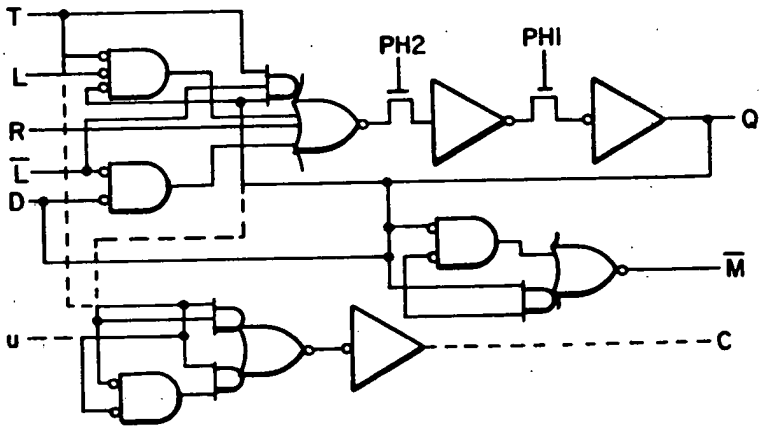


Fig. 19 0(2)

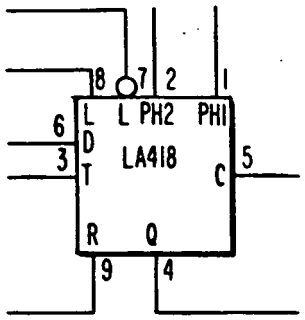


Fig. 19P (1)

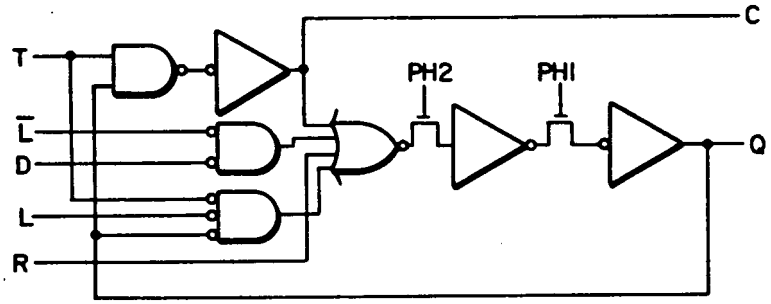


Fig. 19P (2)

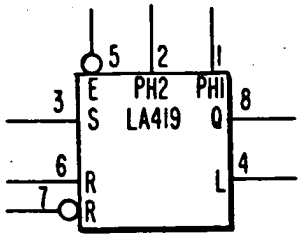


Fig. 19Q(1)

$\overline{En}$	Sn	Rn	Qn	Qn+1	Ln
X	X	X	X	0	0
0	0	X	0	0	0
0	0	0	1	0	1
0	0	1	1	1	0
0	1	0	0	0	1
0	1	X	1	1	1
0	1	1	0	1	0

Fig. 19Q(2)

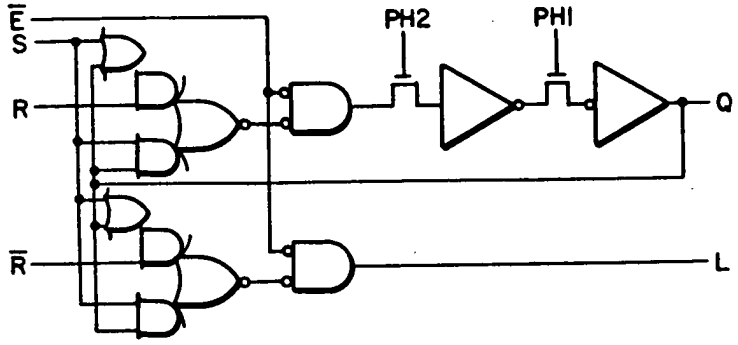


Fig. 19Q(3)

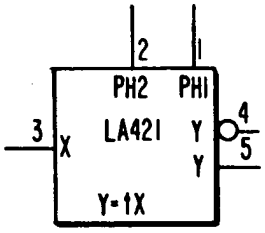


Fig. 19R(1)

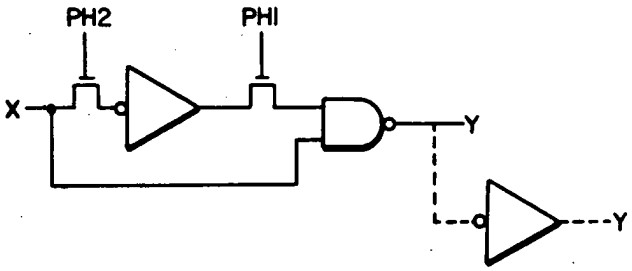


Fig. 19R(2)

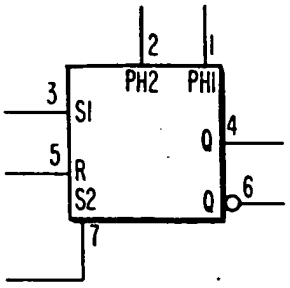


Fig. 19 S (1)

Sn	Rn	Qn	Qn-1
0	X	0	0
X	0	1	1
1	X	X	1
0	1	1	0

Fig. 19 S(2)

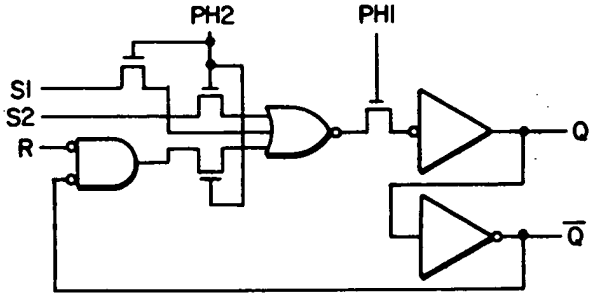
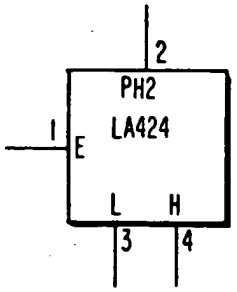


Fig. 19 S (3)

Fig. 19T (1)



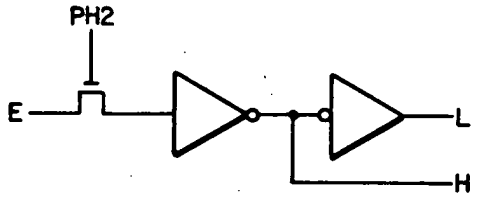
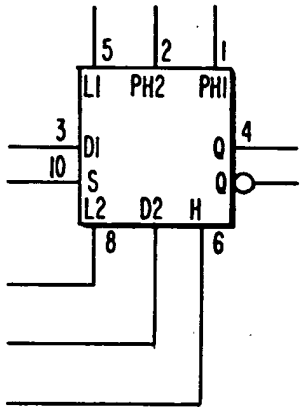


Fig. 19T (2)

Fig. 19U (1)



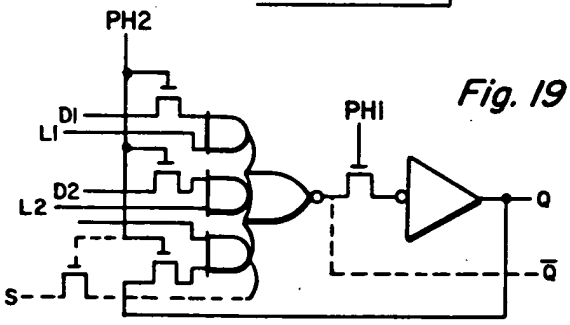


Fig. 19U (2)

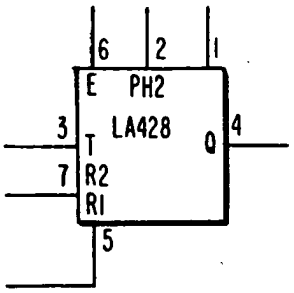


Fig. 19 V (1)

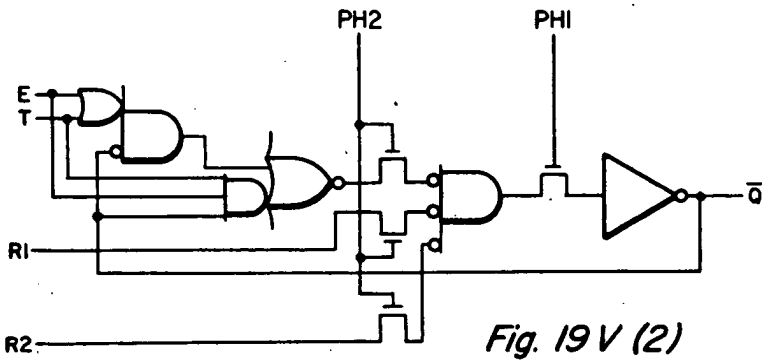


Fig. 19 V (2)

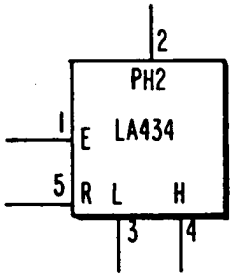


Fig. 19 W (1)

E	R	L	H
0	0	0	1
0	1	0	0
1	X	1	0

Fig. 19 W (2)

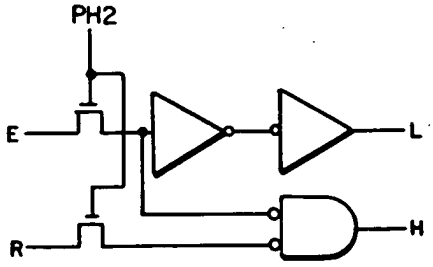


Fig. 19 W (3)

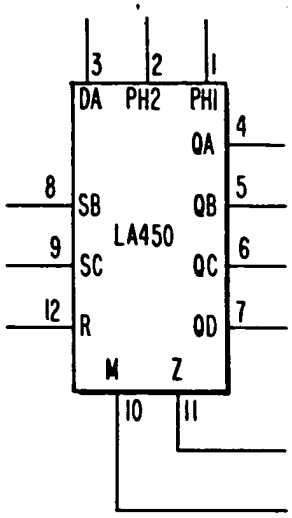


Fig. 19X(1)

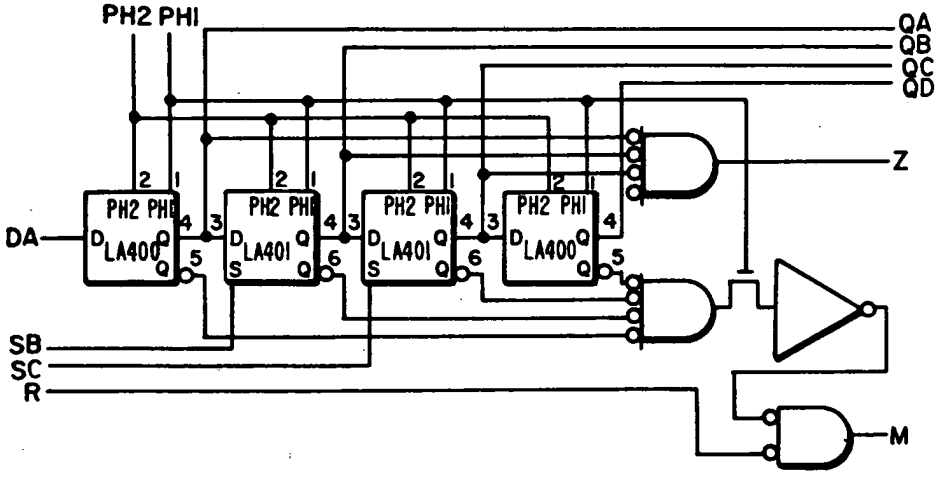


Fig. 19X (2)

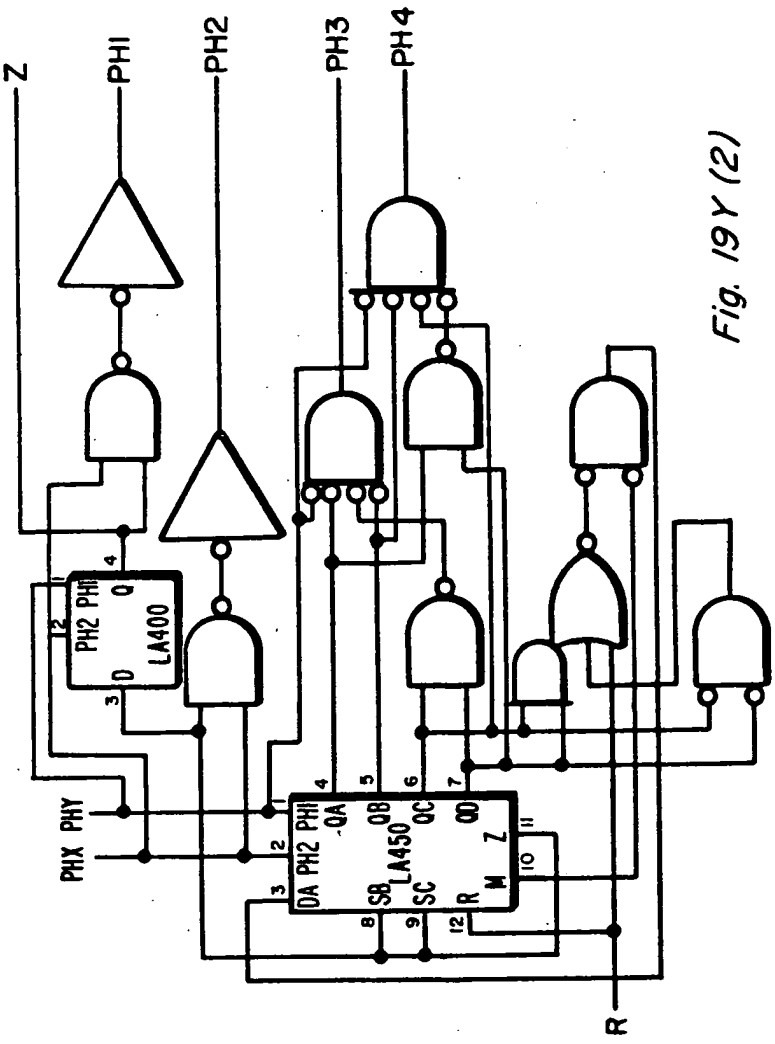


Fig. 19Y(2)

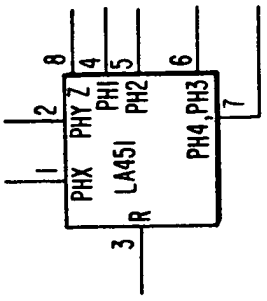


Fig. 19Y(1)

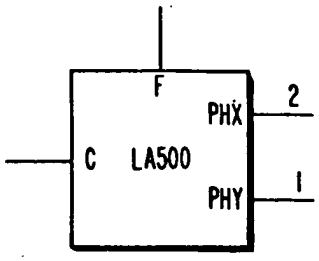


Fig. 19Z (1)

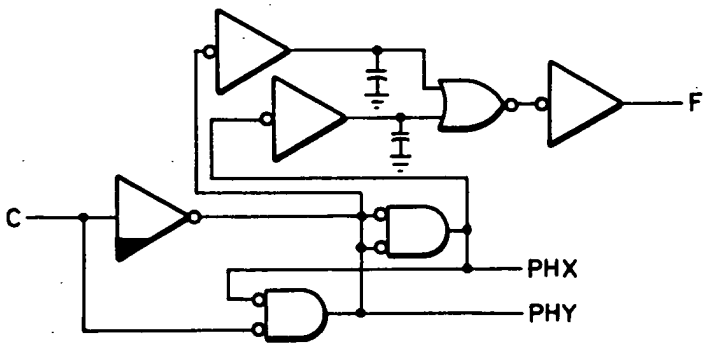


Fig. 19Z (2)

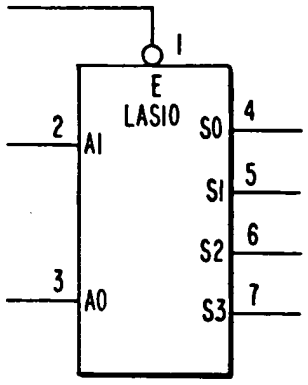


Fig. 19AA (1)

E	AI	AO	SO	SI	S2	S3
0	0	0	1	0	0	0
0	0	1	0	1	0	0
0	1	0	0	0	1	0
0	1	1	0	0	1	1
1	X	X	0	0	0	0

Fig. 19AA (2)

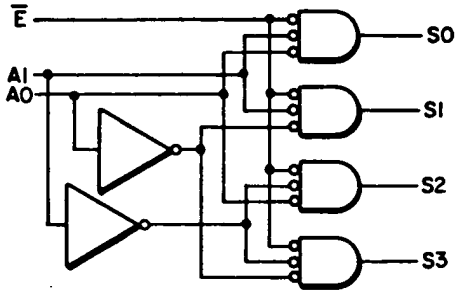


Fig. 19AA (3)

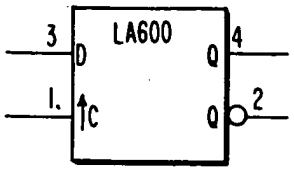


Fig. 19 AB (1)

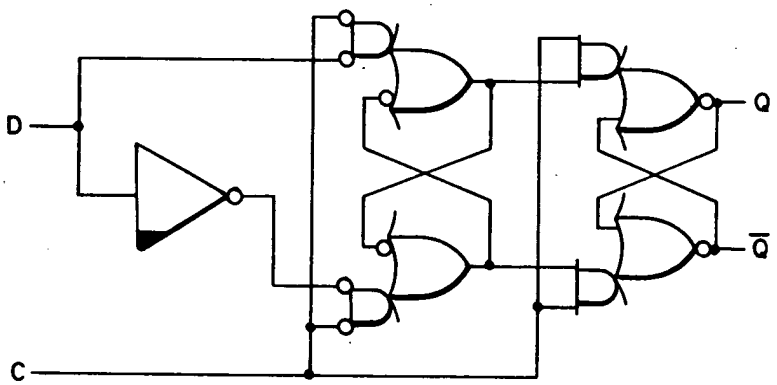


Fig. 19 AB (2)

DOT MATRIX CHARACTER PRINTER CONTROL CIRCUITRY FOR VARIABLE PITCH PRINTING

CROSS REFERENCE TO RELATED APPLICATION

This application is a continuation-in-part of copending U.S. application Ser. No. 955,525, filed Oct. 30, 1978 and entitled Dot Matrix Character Printer With Variable Speed Control, now abandoned.

FIELD OF THE INVENTION

This invention relates to the field of electronically controlled character printers, particularly dot matrix printers which form characters as an array of dots at available locations. More specifically, this invention relates to a control system for such printers to permit the printing, at variable speeds, of characters having different pitch, with the capability of printing characters of different pitch on the same line.

BACKGROUND OF THE INVENTION

Dot matrix character printers are well known in the art, particularly for use as output devices for computers and other electronic devices. These printers are used to produce printed character output on a paper medium in response to receiving electronic signals which correspond to the characters of the available character sets. Individual characters are formed from an appropriate combination of dots selected from an array of available dot locations. Generally, these dot locations are arranged in an n-column by m-row matrix. A printing head contains a solenoid-operated print rod for each row location in the matrix. When the printing of a character requires that a dot be written in a particular row, the corresponding solenoid in the printing head is activated; this causes the associated printing rod to be flung (or "fired") in the direction of the paper (i.e., "target"). An inked printing ribbon is interposed between the printing head and the paper. Thus, the fired printing rod impacts the ribbon and prints an impression of its end surface on the paper. The printing rod is actually a thin wire-shaped element with a flat end having a round cross section. It is this end which forms a dot on the paper.

Typically, the printing head of a dot matrix printer contains printing solenoids. These are assembled into close proximity with each other. The activation of these printing solenoids causes them to heat up. Dissipation of solenoid heat is normally not a problem in slow speed character printers. However, it has been observed to sometimes become a problem in high speed dot matrix character printers. When the print head is activated for a substantial time and the characters to be generated in a relatively short interval require the printing of a somewhat higher than average number of dots during that interval, solenoid heat may be inadequately dissipated. When that happens, solenoid response characteristics may be altered due to parametric variations, and one or more of the printing rods may cause a dot to be printed slightly earlier or later than intended, or even not at all. This degrades the appearance of dots and characters printed under such conditions and may, under severe conditions, even lead to the printing of unrecognizable or incorrect characters. Solenoid power supply failure or degraded performance may also result, as a greater load is imposed thereon. And excessive heat may cause permanent damage to the solenoids or other components. Considerable efficiency is lost if the character printing rate is significantly reduced in order to avoid the solenoid heat dissipation problem, since high reliability would require a substantial reduction in speed.

Typically, in the prior art, in order to change the pitch of printed characters, it has been necessary to provide additional tracks on the encoder at different pitches or to physically change the encoder-to-carriage "gear" ratio. These approaches have typically been cumbersome and not conducive to changing pitch on a character to character basis.

It should also be noted that the printing heads of prior art printers generally have their print rods all arrayed in a single vertical column. Unless the print head is capable of vertical motion, this means that dots formed by adjacent print wires (or rods) cannot overlap; some dot spread caused by the ribbon may, however, allow some blending together of adjacent dots, but resolution is ultimately limited to that provided by tangentially touching dots.

SUMMARY OF THE INVENTION

The above-identified application Ser. No. 955,525 discloses an invention designed to permit the printing of dot-matrix characters of different pitches (i.e., horizontal widths). Character pitch is selectable and variable on a character by character basis, if desired. Electronic signals under the user's control provide for pitch selection. Provision is also made to control the printing speed electronically, to achieve smooth starting and stopping of the carriage while maximizing the character printing rate.

The present invention pertains to circuitry which operates under microprocessor control to provide actuation and control signal for the printer's motors and print head solenoids.

The maximum printing speed is limited by the speed at which the print head solenoids can operate. However, if the text to be printed requires a dot printing rate which would result in overheating of the print solenoids or their power supply, the printing speed is reduced until a safe dot printing rate is achieved. The maximum permissible dot printing rate is a system parameter empirically derived.

Variable rate character printing requires that the printing head carriage traverse the paper at a variable rate. Because the rate of carriage travel is appreciable with respect to the time between a solenoid actuation command and the time the associated printing rod strikes the paper, it is necessary to compensate for variations in the carriage motion in order to always have the printing rods strike the paper in a desired position; this position is, of course, intended to be insensitive to the character printing rate. The situation is analagous to the case of a hunter shooting at a moving duck from a fixed location. In order to allow for the motion of the duck during the time his bullet is in flight, the hunter must aim ahead of the duck; this character printer merely presents the converse situation since the target is fixed and the projectile is fired from a moving platform, although the relative motion problem is the same. Means are therefore provided for deriving a correction factor for actuating the solenoids at the proper times to compensate for the instantaneous rate of carriage motion relative to the paper.

Additionally, in the preferred embodiment of the invention, the print rods are arranged in two columns

such that the print rods in one of the columns are vertically displaced with respect to the print rods in the other column. This permits a higher resolution (i.e., density) dot printing and allows dots to be overlapped a significant amount, so that the dot matrix characters more nearly approximate conventionally printed characters. Separate solenoid firing time signals are required for each column of print rods, to account for their different horizontal positions at any given time.

The present invention also provides such features as the capability of allowing the user to change type fonts and pitch from character to character within the line.

Control responsibility is divided between a microprocessor and dedicated, special purpose control circuiting suitable for implementation on a single large scale integration (LSI) "chip". Hardware control functions and microprocessor firmware control functions are maximally independent. In general, operations which must take place at high speeds are performed in the hardware control circuitry, to permit high speed printing without the need for an expensive, high speed microprocessor or multiple microprocessor.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other features of this invention will become apparent from the following description taken in conjunction with the accompanying drawings in which:

FIG. 1 is a high level functional block diagram of a dot matrix character printer according to the present invention;

FIG. 2 is a diagrammatic illustration of a seven-wire printing head, further depicting one possible mapping of dot buffer output lines to solenoids;

FIG. 3 is a diagrammatic illustration of an eleven-wire printing head, further depicting one possible mapping of dot buffer output lines to solenoids;

FIG. 4 is a functional block diagram of the Printer Control Circuit (PCC) of FIG. 1;

FIG. 5 is a state transition diagram of the Print Enable Circuit of FIG. 4;

FIG. 6 is a state table for the Print Enable Circuit;

FIG. 7 is an overall functional block diagram of the encoder signal processor of FIG. 4;

FIG. 8 is a detailed block diagram of the signal filters of FIG. 7;

FIG. 9 is a detailed block diagram of the transition detector of FIG. 7;

FIG. 10 is a detailed block diagram of the direction decoder of FIG. 7;

FIG. 11 is a detailed block diagram of the transition and increment counter of FIG. 4;

FIG. 12 is a detailed block diagram of the pulse stretcher of FIG. 4;

FIG. 13 is a diagram of the method steps involved in computing the flight time compensation count and for controlling the transition adder;

FIG. 14 is a functional block diagram similar to that of FIG. 1 but further detailing the microprocessor—PCC signal interchange;

FIG. 15 is a modified version of FIG. 4, showing in greater detail the signal paths related to the control of the dot buffers and timers;

FIGS. 16A–16D comprise a detailed apparatus block diagram for the PCC;

FIGS. 17A–17C comprise a cross-reference table relating the blocks of FIGS. 16A–16D to more detailed diagrams in FIGS. 18A–18R;

FIGS. 18A–18AR are circuit diagrams for the blocks in FIGS. 16A–16D, using circuit modules further shown in FIGS. 19A(1)–19AB(2) and

FIGS. 19A(1)–19AB(2) are detailed circuit diagrams for the circuit modules identified in FIGS. 18A–18AR by the designations LAxyz, where xyz is a three digit number.

DESCRIPTION OF AN ILLUSTRATIVE EMBODIMENT

Referring now to FIG. 1, there is shown a high-level functional block diagram of a dot matrix character printer according to the present invention. Paper **12** is the output medium on which the printing is performed. The paper is normally carried on sprockets **14A, 14B** past a print bar not shown, according to conventional practice. The print bar serves the same basic purpose as a platen (i.e., acts as a hard backing surface to print against) except that it does not move the paper. The edges of the paper contain evenly spaced vertical holes which mate with the teeth of drive sprockets **14A** and **14B** for vertically feeding the paper. Images are formed on the paper by a ribbon **16** which may be carried between a pair of ribbon reels **18A** and **18B** or may be a cartridge type ribbon wound on one reel or spool. The print rods or wires from print head **20** force the inked ribbon **16** onto the paper **12** and against the print bar, to cause an image of the print rod to be formed on the paper. Print head **20** is carried on a carriage which traverses the paper for positioning the print head in the proper location as each column of dots is printed. Printing mechanism **22** comprises a mechanical assembly which includes the carriage and the necessary linkages and so forth for moving the carriage, together with a ribbon advancement mechanism for winding the ribbon (e.g., onto reel **18B**) as it is used. The printer mechanism also includes conventional mechanical linkages for connecting the line feed stepper motor **24** to the tractors or drive sprockets **14*b*** and for connecting the carriage motor **26*a*** to the carriage motion mechanism.

A printer control circuit (PCC) **30** provides the actual control signals which cause the carriage to move to the proper location and the appropriate print solenoids to be energized. PCC **30** provides basically three types of control signals as outputs. A line feed control signal is provided as the first output on line **32** to line feed amplifier **34**. In turn, line feed amplifier **34** provides the required drive signal to the line feed stepper motor **24**, to cause the line feed advance mechanism to move the drive sprockets an appropriate distance to advance the paper one line space. A carriage drive control signal is the second PCC output; it is provided on line **36** to carriage motor amplifier **38**. The carriage motor amplifier, in turn, provides the drive signal to carriage motor **26A**, for controlling the motion of the carriage. The carriage motor is also equipped with an incremental shaft position encoder **26B** for providing an output signal on line **42**, to the PCC, representative of the change in angular position of the motor shaft. Since the rotational position of the carriage motor shaft directly corresponds to the position of the carriage and, in particular, the print head, the signal on line **42** represents the position and position changes of the print head. The third output of the PCC is a print head control signal provided on line **44** to the print head driver **46**. The print head driver **46**, responsive to signals provided on line **44**, provides an actuation signal to appropriate sole-

noids in the print head **20** to cause the associated dots to be printed at a desired location.

For use with a computer, or other similar source of data to be printed, the printer is provided with an input interface **48**. This interface provides appropriate input buffering for capturing the data to be printed. A table of character-to-dot patterns stored in ROM **56** of sequential controller **50** is used to convert the character information provided by the input data source into dot matrix information representing each character to be printed. By varying the character-to-dot pattern mapping, different fonts (such as different typefaces or typefaces for different alphabets—e.g., Cyrillic, Greek, etc.) may be printed. Characters having the same basic font (i.e., typeface) may be spread or compressed in the horizontal dimension by altering the spaces between dots and the blank spaces between characters, thereby altering the pitch of the characters. And, since the mapping must be done on a character-by-character basis, it is possible to change character size and font on the same basis. There is no provision for changing the size of printed characters in the vertical dimension if the number of vertical print rod positions would thereby be exceeded, but various character widths can be accommodated, and vertically compressed or stretched characters can be generated using less than, or at most, the full number of vertically available print rod positions.

A sequential controller **50** for generating PCC control or command signals and provided character-to-dot mapping comprises a processor unit **52**, random access memory (RAM) **54** and a read only memory (ROM) **56**. Interface **48**, processor **52**, RAM **54**, ROM **56** and PCC **30** are all interconnected via a bus or communication link **58**.

Character information to be printed is fed from the input interface **48**, via bus **58**, to RAM **54**, where it is temporarily stored. ROM **56** controls the operation of the processor **52** and the data paths between the various elements connected to the bus **58**. Using the character-dot mapping provided by the ROM, processor **52** provides instructions to PCC **30** regarding the dot pattern to be printed. In turn, PCC **30** determines the appropriate times, as a function of carriage position, for activating the print head driver to fire the print head solenoids.

While various components may be used to fabricate the box described above, it is particularly noted that the models 8080A and 8085 microprocessors manufactured by Intel Corporation have been found suitable for the processor **52**. Accordingly, the particular embodiment disclosed in detail herein is described with reference to the use of the aforesaid Model 8080A. The printer control circuit **30** could also be implemented in a number of ways. However, in order to minimize cost and number of interconnections, we have implemented it as a single large scale integrated circuit using metal oxide semiconductor technology, with a design architecture suitable for control by a microprocessor.

The printer control circuit is capable of operating in two modes for generating characters of different pitches. The first mode permits the use of a conventional seven wire print head containing seven print rods arranged in a single vertical column. The second mode is unique to this invention and permits use of two, horizontally displaced columns of print rods. The specific illustration of this mode, described below, shows the use of an eleven wire print head having a first vertical column of six print rods and a second vertical column of five print rods. The embodiment illustrated further shows the rod positions in the second row vertically spaced from the rod positions in the first row so as to achieve a vertical interleaving of positions and permit the printed dots formed by vertically adjacent rods in the two rows to overlap. FIG. **2** illustrates the print rod positions in a seven wire head and FIG. **3** illustrates the print rod positions in an eleven wire head. A functional block diagram for the printer control circuit (PCC) **30** is shown in FIG. **4**, to which reference will now be made. Character data to be printed and supervisory commands to the PCC are received via the bus **58** which enables the PCC to communicate with the sequential controller and input interface, as required. Information received by the PCC upon the bus **58** is captured by an input buffer **62**. The input buffer **62** feeds this information to the various functional elements of the PCC via a parallel bus **64**. In the drawing, bus **64** is indicated separately at the appropriate functional blocks (by an arrow comprising two spaced, short parallel lines terminated by an arrowhead at one end and broken off at the opposite end); it should be understood that the bus interconnects all of the indicated blocks even though the full interconnection is not illustrated.

One of the elements connected to bus **64** is a command decoder **66**. This command decoder is also connected via lines **68** to the various other elements of the PCC; such connections are not explicitly shown, however, in order to maintain clarity in the drawing. Command decoder **66** detects and decodes the commands provided by the sequential controller and through the use of appropriate steering logic activates, deactivates and controls the functional blocks shown in FIG. **4** to achieve the operation directed by the commands.

As briefly stated above, the function of the PCC is to supply appropriate signals to drive the line feed amplifier **34**, carriage motor amplifier **38** and print head driver **46**. The dominant function of the PCC is to implement the required control of the print head driver **46** for printing, with variable pitch and performing the "duck shooting" operation involved in printing with a variable carriage speed. The desired carriage speed is a function of carriage position in the line, with the objective being to maintain maximum throughput of characters within the limits imposed by available acceleration, start/stop dynamics of the print head, and solenoid actuation rates which avoid excessive heating. The processor **52** determines the appropriate carriage speed and the PCC actuates the print head driver accordingly. Thus, the basic information required by the processor **52** is the position and velocity of the print head. This information is obtained from encoder **26B** which is driven by the carriage motor **26A**. Encoder **26B** is an incremental two-channel encoder which provides a digital output. A short up-down position counter **72** receives the output of the incremental encoder **26B** via an encoder signal processor **74** which suitably conditions and formats the output of the encoder. Thus, the position counter **72** keeps track of the relative position of the print head, modulo its radix. The output of position counter **72** is supplied to the output buffer **78** on demand, and, from the output buffer, to bus **58** and processor **52**. Processor **52** periodically examines the position information provided by the position counter **72** and calculates print head velocity from the change in position during the inter-sample time. In the particular embodiment heretofore implemented, the processor **52** samples the output of the position counter approximately once every 2.5 ms. The change in position ob-

served at each sample time is summed in a position register (i.e., address location) in RAM **54**. Absolute position information is obtained by initializing the position register to a known value when the carriage is forced against a fixed stop.

The initialization procedure of the printer requires that the carriage motor be commanded to drive the carriage in the direction of the stop. When the processor detects no change in the position counter's output during a predetermined number of sample times in which the carriage is being commanded to move toward the stop, it "knows" that the carriage has already reached the stop; the position register is then initialized. All relative motion can then be compared against this initial position, to be transformed to absolute position.

While various encoder designs well suited to this application will be apparent to those skilled in the art, this system has been successfully implemented with an encoder of the type which uses two tracks to provide a pair of (phase) quadrature output signals; these encoder signals are supplied to the PCC encoder signal processor (ESP) **74** via lines **42A** and **42B** (which are collectively shown as line **42** in FIG. **1**). Encoder signal processor **74** is used to give "clean" position decoding and insure that only valid state changes on line **42A** and **42B** are interpreted by the PCC as indicative of carriage motion. ESP **74** may, for example, comprise a so-called "majority vote" circuit which uses a high sampling rate to distinguish between valid and invalid state changes in the encoder's quadrature output signals.

At this point it would be helpful to the further understanding of the operation of the PCC to digress briefly for an explanation of character "cell" structure. All printing of characters by this system takes place within programmable character cells whose fundamental units are encoder "transitions" (i.e., logic state changes in the encoder quadrature output signals). As successfully implemented, for example, each encoder transition may correspond to print head or carriage motion of 1/660 of an inch. A character cell consists of two sections, the first (from the left) is the increment field in which dots may be printed; the second is the inter-character space. Within the increment field, two or more increments are equally spaced. Their spacing is a programmable parameter called "transitions per increment," TPI. TPI is often also referred to as the radix of the character and is a function of character width or pitch. The number of increments per character, IPC, is also programmable. Since an increment occurs at both the beginning and end of the increment field, the width of the increment field (in transitions) is given by $TPI \times (IPC-1)$. The inter-character space has no fine structure and is directly programmed as transitions per space, TPS. As the name implies, dot printing is inhibited (i.e., no increments occur) in the inter-character space. A character cell's width is equal to the sum of the widths of the increment field and the inter-character space, so that total width is given by:

$$\text{WIDTH} \quad TPI \times (IPC-1) + TPS.$$

The diagram below illustrates a single cell of width **40**, IPC **13**, TPI **3**, and TPS **4**:

I..I..I..I..I..I..I..I..I..I..I..I..I..Isss

where I represents an increment, s represents a space, and the "." symbol represents any other transition.

Each of the parameters mentioned above—TPI, IPC and TPS—can be programmed into the PCC from the processor **52** to select a particular font compression or pitch. While printing, the PCC counts transitions and increments so that it knows when a character boundary, increment or space is occurring.

The counting of transitions and increments is done by a transition and increment counter **86**, which includes a transition counter for counting transitions and an increment counter for counting increments (see FIG. **11**). Each of these counters is a programmable radix upcounter which provides an overflow signal as its output. The overflow signal occurs on the arrival of the next counting pulse after (i.e., the stored count equals the counter's radix).

When the printing of a character is begun, the transition counter uses the value of TPI as a radix. Every time the transition counter overflows, an increment is generated which, among other things, causes the increment counter to change its count. When the increment counter reaches a count equal to IPC, the radix of the transition counter is changed to TPS, so that the proper inter-character space is generated. When the next increment is received, the increment counter goes to a value less than IPC (either 0 or IPC−1, depending on direction). At this time, a character boundary signal is generated and the radix of the increment counter again becomes TPI.

It can be seen that the transitions per space could be any non-zero number. However, the fact that the timers operate on every increment makes it desirable for TPC to be at least equal to the value of TPI. The counters involved can be initialized by a command from the sequential controller **50**. Table 1, below lists several character pitches and the parameters that generate them.

TABLE 1

CPI	TPC	IPC	TPI	TPS
10	66	11	6	6
12	55	11	5	5
13.2	50	10	5	5
16.5	40	10	4	4
5	132	11	12	12
6	110	11	10	10
6.6	100	10	10	10
8.25	80	10	8	8

Where CPI means characters per inch and TPC means transitions per character (i.e., pitch). The first four rows represent normal pitches and the second four rows represent double width pitches.

Encoder signal processor **74** provides three output signals derived from the quadrature signals received from the shaft encoder. The first of these signals is a direction signal, on line **82**, indicative of the direction of carriage motion. The direction signal is provided to position counter **72** in order to cause it to count either up or down, as appropriate. Carriage motion in a first direction will cause the position counter to count up and carriage motion in the other direction will cause the position counter to count down. A second output of the ESP, provided on line **84**, is an encoder transition pulse signal. The latter signal comprises a pulse each time one of the two quadrature encoder signals on lines **42A** and **42B** changes state, indicating that the carriage has moved one transition distance from its prior position.

The pulses in the encoder transition pulse signal on line **84** are supplied to and counted by the position counter **72** for keeping track of changes in carriage position. Lines **82** and **84** are also connected to transition and increment counter (T&IC) **86** and line **82** is further connected to a pulse stretcher **88**. The third output signal from the ESP **74** is a speed transition pulse signal on line **92**; this signal comprises the second input signal to the pulse stretcher **88**.

To implement the above-described performance capabilities, print commands must be given to the solenoids at varying (i.e., not fixed) positions relative to the "target" location at which printing is desired; the solenoid actuation (i.e., "firing") positions are functionally related to print head velocity, since the distance travelled by the print wires is fixed. Sequential controller **50** computes the required solenoid firing position and PCC **30** generates the actual signals which control solenoid actuation in accordance with the calculations of the sequential controller. The transition and increment counter **86** is the starting point within the PCC for the generation of the necessary control signals. T&IC **86**, as its name implies, comprises a pair of counters for respectively tracking transitions and increments. Signal inputs for the T&IC **86** are the direction signal on line **82** and the encoder transition pulse signal on line **84**. Since the carriage is moving while the printing operation takes place, the print rods must be "fired" from the moving print head before the carriage reaches the impact location. Normally, the mean value of this lead "time" is known for an average carriage speed, one of the system design parameters. However, carriage speed is variable in this invention and other, non-average speeds will require a greater or smaller lead (i.e., positional displacement) between print wire firing and impact, depending upon whether the speed is below or above average. Translating the lead into position units, there is a nominally known displacement (in transitions) between the impact position on the paper and the position of the print head when the solenoids are actuated. Appropriate circuitry is provided for reading the "position" indicated by the T&IC and triggering the firing of the solenoids when the position of the carriage indicated by the T&IC is the correct distance from the target position on the paper, for the then current carriage velocity. Thus, a preselected transition count (e.g., a count equal to the transition counter radix) in the T&IC comprises a trigger signal for the printing solenoids. The solenoids are actuated in response to the appearance of this count. Processor **52** supplies a transition add/subtract command to the T&IC via bus **64** to cause the transition count stored therein to be either increased or decreased. This creates a false or pseudo-transition reading in the T&IC; this reading may be used to actuate the printing solenoids by forcing the "triggering" count to appear at the position suggested by print head velocity. Thus, the transition count indicated in the T&IC to is shifted from the count registered in position counter **72** by an amount representing a flight time compensation count. That is, this difference represents the displacement, in transitions, between the point of solenoid actuation and the point of impact.

The transition counter portion of the T&IC **86** comprises a programmable radix counter, the radix of which may be set to correspond to either the parameter TPI or the parameter TPS. The increment counter portion of the T&IC comprises a counter having a single programmable radix corresponding to the parameter IPC. The radices are supplied from the sequential controller **50** via bus **64**.

The transition and increment counter **86** provides three output signals. The first of these is a character boundary signal provided on line **94** to the print enable circuit (PEC) **96**. The character boundary signal merely defers the execution of the print-start command until the print wire input position enters the cell of the character to be printed. The second signal provided by the T&IC, on line **98**, is a net transition signal. Line **98** is connected to increment shift register **100** for supplying the net transition signal thereto. The third output signal provided by the T&IC, on line **102**, is a primary increment signal. The primary increment signal is supplied to increment shift register **100** and to increment steering circuit **104**.

The function of print enable circuit **96** is to provide internal PCC control (i.e., enablement and disablement) of the elements used to trigger the solenoids. Basically, the print enable circuit is a finite state machine the operation of which is defined in FIGS. **5** and **6**. Remaining briefly with FIG. **4**, however, it will be seen that the PEC **96** has three signal inputs and one signal output. The first signal input is the character boundary signal on line **94**. The second signal input is a printing done signal on line **106** from buffer monitor circuit **108**. Sequential controller **50** provides the third input signal, a print start signal, via bus **64**. The output of PEC **96** is a print enable signal on line **112**. The print enable signal is supplied to increment shift register **100** and to increment steering circuit **104**. The print enable signal will assume a first logic state when printing is to be enabled and will assume a second logic state when printing is to be disabled. Referring now to FIG. **5**, there is shown a state transition diagram illustrating the normal operational sequence of the print enable circuit. The PEC has three states, labeled respectively, **114**, **115**, and **116**. Printing is disabled in states **114** and **115**; only in state in **116** is printing enabled. Assuming initially that the PEC is in state **114**, as it would be after printing has previously been terminated or at a time following initialization but prior to the start of any printing, a print start command from the sequence controller received via bus **64**, will cause a change to state **115**; at this point the PEC assumes control over the printing operation. Once the PEC is in state **115**, it monitors the character boundary signal on line **94**; when that signal indicates that the leading edge of the character cell boundary has been crossed, the PEC advances to its third state, **116**, and provides the print enable signal on line **112** in the state which permits printing. The printing operation will continue until buffer monitor circuit **108** provides a printing done signal on line **106** to indicate that there is no more information to be printed. Upon receipt of the printing done signal, the PEC will return to state **114** and disable printing by changing the state of the print enable signal on line **112**. Printing will not resume until another print start command is received from the sequence controller. Thus, the state **114** may be called an idle state, state **115** corresponds to a state of waiting for a character cell boundary and state **116** corresponds to the active printing state. The state transition table shown in FIG. **6** provides another description of the PEC, assuming a particular implementation in which the printing disabled condition corresponds to a logical "0" print enable signal and a logical "1" print enable signal permits printing to be performed. Corresponding assumptions are made with respect to the logic values of

the PEC input signals. The symbol "X" is used in FIG. 6 to indicate, with respect to input, that either logic level may exist and, with respect to states, that any state satifies the condition.

It is noted that in the above description the term "character boundary" corresponds to the increment which makes the increment counter in the T&IC exit the intercharacter space. The term "increment" refers to that transition which makes the transition counter in the T&IC overflow; that is, count up from maximum to zero or down from zero to maximum. And "intercharacter space" is that interval during which the increment counter value is equal to its radix, (i.e., its maximum value).

Two dot buffer and timer circuits, **120A** and **120B** are provided. When a two-column (e.g., eleven-wire) print head is employed, one of the dot buffer and timers is associated with a first of the columns and the other dot buffer and timer is associated with the other of the columns. These might correspond, respectively, to five and six dot columns, respectively. When only one column of print rods is provided in the print head, such as for a seven-wire head, both dot buffer and timer circuits are operated in parallel. The dot buffer and timer circuits receive character dot data from the processor and fire the corresponding print solenoids at appropriate times. Each dot buffer and timer circuit includes a plurality of registers, a register controller, an output buffer and a timer. The registers are preferably arranged as a first-in-first-out (or fifo) stack buffer. One of the registers in the stack is an input register which is loaded by the processor; another register comprises an output register. Data propagates through each register, beginning with the input register, until it eventually reaches the final, output register. The output of the output register is gated to the print head amplifiers at a precise point in the head travel and is held for a precise amount of time, as controlled by a timer in the dot buffer and timer circuit. The purpose of the dot buffer and timer circuits is to allow the processor **52** to supply dot data for the print head asynchronously with respect to the time that such data is needed for actuation of the solenoids. The controller in the dot buffer and timer circuits consists of a memory which tracks the presence, in each register, of data to be printed together with logic which causes all data to advance as far as possible through the fifo stack. The controller detects when the timer has ceased actuating the output buffer and thereupon advances the data in the registers. This creates space at the input of the stack. A buffer status signal is supplied on lines **122A** and **122B** to buffer monitor circuit **108**. When a predetermined number of spaces exist in the fifo stack(s), as indicated by the buffer stack signals to the buffer monitor, the buffer monitor generates a dot data request signal to the processor, on line **124**. The processor then supplies more data into the dot buffer input register. When both buffer stacks are completely empty, the buffer monitor generates a printing done signal on line **106**, to reset the PEC **96**.

The timer in each of the dot buffer and timer circuits is used to cause the dot data to be presented to the printing element (i.e., solenoid) for a programmable amount of time determined by the characteristics of the element being controlled. The timer consists of a binary counter which counts a programmable number of clock cycles. It starts counting on command of the increment steering circuit **104**, unless already running. Each time the timer halts, it supplies a signal to the stack controller to allow new data to be presented to the output buffer.

The output buffer in the dot buffer and timer circuit allows the output of the stack to propagate to the print head whenever the timer is running.

Each register in the fifo stack in the dot buffer and timer circuits **120A** and **120B** contains storage sufficient for one vertical column of dots. In the particular implementation shown in the drawings and discussed herein, one of the dot buffers contains five-bit registers and the other contains six-bit registers, thus providing for the control of a maximum of eleven printing solenoids in the print head. It is readily possible, of course, if only a single column print head is to be employed, to use but one dot buffer and timer circuit having the same number of bits per register as the number of print wires. The increment shift register **100** and increment steering circuit **104** also become unnecessary in the latter application, as the single dot buffer and timer circuit is directly controllable by the primary increment signal, explained below.

With a two column printing head, solenoid control is complicated by the fact that the two columns of wires are physically displaced, or separated from each other, in the horizontal direction. Therefore, in order to print a single vertical line of dots, the trailing set (i.e., column) of wires must be fired a number of transitions later than the leading set—the number of transitions corresponding to the displacement between the two columns of wires, measured in transitions. Thus, the two columns of wires are provided with two different firing control signals. These are referred to as primary and secondary increment signals. The primary increment signal fires the leading set of wires and the secondary increment signal fires the trailing set of wires. When printing, the secondary increment repeats the primary increment pattern, only delayed by the necessary number of transitions. It should further be noted that printing may be done either left to right or right to left. In one direction, one of the columns will be the leading set and the other will trail; but the situation will be reversed in the opposite direction, with the other column being the leading set.

The primary increment signal is generated by the transition and increment counter **86** and is provided on line **102** to the increment shift register **100** and increment steering circuit **104**. Increment shift register **100** performs the transition delay function for generating the secondary increment signal therefrom. The secondary increment signal is provided on line **126** to increment steering circuit **104**. Net transition signal **98** is essentially used as a clocking signal for increment shift register **100**.

When the printer is operated in the two column print head mode, the increment steering circuit directs the primary increment signal to the dot buffer and timer for the leading set of print wires and the secondary increment signal to the dot buffer and timer for the trailing set of print wires. When the printer is operated in the single column print wire mode, increment steering circuit **104** directs the primary increment signal to both dot buffer and timer circuits, as the secondary increment signal is not used.

The PCC also drives the carriage motor amplifier in the carriage servomotor system, for controlling carriage motor speed. Carriage motor speed control is accomplished by a servo system. Processor **52** determines the maximum possible carriage speed based upon

the highest solenoid actuation rate at which the print solenoids will operate properly and which may be maintained without causing overheating. It provides a speed command signal via bus **64** to bit rate multiplier (BRM) **132**. The BRM is of conventional design and provides two output signals. The first of these is a command sign signal, on line **134**, to indicate the direction in which the motor is being commanded to operate; the second of the BRM output signals is a command pulse signal on line **136**, which provides the actual motor velocity information. The speed transition pulse signal on line **92** and the direction signal on line **82** comprise corresponding signals indicative of the actual motion of the carriage motor. These latter two signals are received by pulse stretcher **88** which, in turn, supplies the feedback signals required in the servo loop. A feedback sign signal is provided on line **138** and a feedback pulse signal is provided on line **139**. The speed steering circuit **142** connects lines **136** and **139** to the appropriate output pins **144** and **146** in accordance with the associated sign signals carried on lines **134** and **138** respectively. These pulse trains are received by the carriage motor amplifier **38** which integrates these pulse signals to obtain their average values. The carriage motor amplifier is the "error" amplifier of the carriage motor servo; it determines the difference between the commanded speed and direction and the actual speed and direction of the motor and provides a drive signal to the carriage motor to drive the carriage motor toward the commanded speed.

The command given to the servo mechanism by the BRM consists of a stream of narrow pulses. The pulses in the speed transition pulse signal on line **92**, containing the motor speed information used for feedback, have the same width as the BRM command pulses; however, there are several, e.g., eight, command pulses for every speed transition pulse provided on line **92**. The servo mechaism is designed, typically, to respond to the average value of the command pulse the feedback pulse signals; thus, it becomes necessary to "stretch" the speed transition pulses a commensurate amount. This is done by producing for every speed transition pulse on line **92**, one feedback pulse on line **139**, which feedback pulse is eight times as long as the speed transition pulse. For example, if the command pulses are each 6.5 ms in width, the feedback pulses are approximately 52 ms in width.

In FIGS. **7-12**, there is a further breakdown, still on the block diagram level, of some more of the blocks shown in FIG. **4**. FIGS. **7-10** provide a breakdown of the encoder signal processor **74**. An overall block diagram of the encoder signal processor is shown in FIG. 7. The purpose of the encoder signal processor is to convert quadrature encoded squarewave position signals, received on lines **42A** and **42B** from the two encoder tracks, into transition event and direction signals that can be counted in up/down counters and used to trigger transition-related events. The encoder signal processor which performs these functions consists of a direction decoder **202** and, for each of the two channels of encoder signal input, a signal filter, **204A** or **204B**, and an associated transition detector **206A** and **206B**. The outputs of the transition detectors on lines **208A** and **208B**, respectively, are combined together by OR gate **210** to provide the encoded transition pulse signal which is supplied to the position counter **72** and T&IC **86** on line **84**. The speed transition pulse signal is provided on line **92** by AND gate **212** which receives as its inputs the transition detector output signal on line **208A** from one of the channels and the input of the transition detector **206B** of the other channel, on line **214**. Direction decoder **202** receives the outputs of the two signal filters on lines **214** and **216** to provide the direction signal on line **82**. It will thus be seen that the encoder transition pulse signal provides a pulse each time there is a state transition in either of the two signals provided by the encoder, and that the speed transition pulse signal provides a pulse when there is a transition on one of the channels while the other channel is in the "1" state. In steady state operation, a speed transition pulse will appear on line **92** every fourth transition. In the specific implementation discussed herein, the width of these pulse signals corresponds to one clock period of the system clock.

The output of the direction decoder is, by contrast, a level which represents the sign (+ or −) of the transition and identifies which of the two quadrature signals leads in phase.

The purpose of signal filters **204A** and **204B** is to remove short duration noise pulses from the signals on line **42A** and **42B** so that they do not cause false outputs and false responses by the transition detectors and direction decoder. It has been found particularly desirable to implement the signal filters as majority vote circuits which compare two new samples of their input with the previous output. The outputs of the signal filters are allowed to change state only if the two new samples agree with each other. For further reliability, an even greater number of samples could be used. The two samples of the input of the signal filter are taken at a normal system clock rate but approximately 180° out of phase with each other, thereby yielding a net sampling rate which is twice the normal clock rate.

An appropriate block diagram for each of the signal filter elements **204A**, **204B** is shown in FIG. **8**. While the example is described in terms of signal filter **204A**, it is equally applicable to signal filter **204B**, of course. Four conventional sample-and-hold (S & H) circuits and a majority vote circuit are employed in each signal filter. The sample-and hold-circuit **222** provides the output of the signal filter which represents the signal provided by the majority vote circuit **224** at sampling time **T1**. The two signal samples are taken at times **T2** and **T3**, respectively, by S & H circuits **226** and **229**. The fourth sample and hold circuit **232**, samples the prior output which is present on line **216** at time T2, as well. As explained above, sample times T2 and T3 are displaced from each other by approximately 180° but occur at the same rate, being derived from the same clock. Majority vote circuit **224** reads the outputs of S & H circuits **226**, **229** and **232** appearing on lines **227**, **229**, and **231**, respectively, and provides an output on line **225** corresponding to the state of the majority of its inputs. The signal input on line **42A** is preferably synchronized so as to be stable at the T2 and T3 sample times. A clocked D-type flip-flop, Schmitt trigger or similar device (not shown) may be used for this purpose.

The purpose of the transition detectors **206A** and **206B** is to produce an event pulse each time their input signals change state. This is accomplished, for example, with the arrangment shown in FIG. **9**. As shown therein, the transition detector **206A**, for example, consists of a delay element **236** and an exclusive-OR gate **238**. The output of the delay element, on line **237** and the input, on line **216**, provide two inputs of the exclusive OR gate **238**. This produces at the output of the

exclusive OR gate, on line 208A, a stream of pulses each of whose width is equal to the delay time of the delay element. The delay time of one clock period of the system clock is satisfactory for this purpose. Thus, delay element 236 is shown as also receiving the system clock on line 239.

The output of the direction decoder 202 in the encoder signal processor accompanies each of the transition event signals (i.e., the speed transition pulse signal and the encoder transition pulse signal) to their destinations. The purpose of the direction decoder is to decode the quadrature encoded direction information in the two input signals on lines 42A and 42B, producing an output signal whose states correspond to the directions associated with the transition event pulses. One satisfactory exemplary implementation of the direction decoder is shown in FIG. 10. As illustrated therein, the direction decoder 202 consists of a delay element 242 and then exclusive-OR gate 244. One of the inputs, such as the one on line 216, is delayed by the delay element 242 and the delayed signal is provided on line 243 to one of the inputs of the exclusive-OR gate 244. The other input, such as the one on line 14, is supplied directly to the other input of the exclusive-OR gate. The output on line 82, if observed only during the periods when the encoder transition event signal is asserted, will (at those times) be in one state when the transition was in one direction and then the other state when the transition was in the opposite direction. Delay element 242, like delay element 236 provides a single clock period delay and receives a system clock as an input on line 246. It will be observed that if the clocking signals on lines 246 and 239 are the same, the outputs of the delay elements 236 and 242 are the same. Hence, a single delay element may be shared between the direction decoder and one of the transition detectors.

Referring to FIG. 11, there is shown a block diagram breakdown of the T&IC 86. The transition and increment counter 86 has three component elements: transition adder 252, transition counter 254 and increment counter 256. The transition counter and increment counter have already been alluded to above. The transition adder 252 basically operates under the control of the sequential controller 50 to provide the required offset in the count maintained in the transition counter from the count maintained in the position counter (i.e., the flight time compensation count) so as to cause the transition counter to produce the solenoid-actuating count at the appropriate location. When a two-column print head is used, the transition adder also "moves" the primary increment when the direction of printing changes, to account for the displacement between the two columns of print wires. Additionally, the transition adder serves to eliminate single transitions having opposite signs (i.e., direction) from the immediately preceding and following transitions, such as are produced by the operation of the transition add/subtract circuit. Other such signal transitions should not and would not occur but for the presence of "noise" and sampling inaccuracies.

Included within the transition adder 252 is a net transition generator 252A which actually provides the two transition adder output signals, the net transition signal (on line 98) and the net direction signal (on line 253). In addition to eliminating pairs of single transitions of opposite sign, the net transition generator 252A operates under the command of the sequential controller 50 to either add transitions on line 98 which are in addition to those provided on line 84, or to prevent pulses on line 84 from reaching line 98, in order to force the transition counter 254 to reflect properly the positional correction required for actuating the solenoids at the proper time. The net direction signal on line 253 is simply the direction signal which corresponds to the net transition signal on line 98; it is the direction signal on line 82 as modified by the operation of the net transition generator 252A in adding, subtracting or eliminating pulses.

Transition counter 254 comprises a programmable radix counter which may select its radix from two possible parameters, TPI and TPS, the values of which are set by the sequential controller 50 via the bus 64. Transition counter 254 is an up/down counter which receives counting direction control from the net direction signal on line 253 and which counts the pulses in the net transition signal on line 98. The output of the up/down counter is decoded to provide as the output of the transition counter a signal, termed the primary increment signal, on line 102 when the transition count of the up/down counter corresponds to the position at which the printing solenoids should be actuated. Increment counter 256 receives the primary increment signal and the net direction signal as inputs and provides the character boundary signal as an output, on line 94. The increment counter is a simple, programmable radix counter whose radix, corresponding to the parameter IPC, is set by the sequential controller via the bus.

The third block shown on FIG. 4 to be broken down for further explanation of its structure is the pulse stretcher 88, shown in FIG. 12. The pulse stretcher comprises two components: speed transition processor (STP) 262 and pulse generator 264. The speed transition processor receives the direction signal on line 82 and the speed transition signal on line 92 as well as a 19.2 kHz pulse on line 266; its outputs are the feedback sign signal on line 138 and a feedback pending signal on line 268. The feedback pending signal comprises one of the inputs to pulse generator 264. The 19.2 kHz pulse signal on line 266 is the other input to the pulse generator, and the feedback pulse signal on line 139 is its output. The STP contains a one-bit memory of feedback sign and one-bit memory of feedback pending. If a speed transition occurs with a direction (i.e., sign) which matches the previously established feedback sign, the feedback pending signal is then asserted. The feedback pending signal is deasserted immediately after the next occurence of the 19.2 kHz pulse. If the direction of the speed transition pulse does not match the feedback sign, the feedback sign is changed, but no feedback pending signal is generated. The pulse generator 264 samples the feedback pending signal on line 268 each time a 19.2 KHz pulse occurs and asserts a feedback pulse on line 139 appropriately, thus producing pulses approximately 52 microseconds wide, eight times the width of the command pulses on line 136 from the bit rate multiplier 132.

As stated above, the sequential controller 50 is the commanding agent, or "brain," driving the PCC 30. Among other things, the sequential controller determines the flight time compensation count which is necessary and provides transition add/subtract commands to the transition adder to cause the net transition generator to feed a lesser or greater number of pulses to the transition counter, to produce the correct flight time compensation count. The sequential controller drives the carriage motor to the highest available speed consistent with satisfactory acceleration and solenoid opera-

tion. The upper limit on carriage motor speed is partially a function of the dot density of characters to be printed over a preselected time interval, so that the solendoid actuation rate does not exceed empirically derived bounds for satisfactory operation. Accordingly, the carriage speed is greater in blank regions, as the solenoid actuation rate does not impose a limit at all in those areas.

In the present implementation, the system clock speed of 2 MHz has been selected because it allows operation of the model 8080A microprocessor at nearly its maximum speed and when divided by 13, yields 153.6 kHz, to provide the 6.5 microsecond sampling interval used by the encoder signal processor **74**. It also provides the clock signal needed by input interface **48** for serial data communication; in particular, it permits the use of many standard signaling rates such as 9600 bits per second and derivatives thereof. The same 153.6 kHz signal drives BRM **132**. As a byproduct, the BRM provides a signal at 1/128 this frequency which, when divided by 3, provides the 2.5 ms sampling interval used by sequential controller **50** to process the readings of position counter **72**.

While the method employed by the sequential controller for flight time compensation may be generalized to the extent illustrated in FIG. **13**, the flight time compensation count needed by a particular embodiment of this invention is a function of carriage velocity; if carriage acceleration is not kept low enough, the compensation function must depend on that parameter too, of course. The compensation available is limited by the velocity range resolved by minimally incrementing or decrementing the flight time compensation count. Since the compensation required is dependent on the mechanical parameters of the printer (i.e., the goal is to compensate for the finite flight time of print wires and other dynamically induced positional offsets), the actual compensation function needed is characteristic of the particular printer. The appropriate printer parameters may be obtained either by modeling or by empirical evaluation.

An illustration of the sequential method steps required for flight time compensation count derivation is shown in FIG. **13**. The first step, **310**, is to measure the positional change of the carriage which occurred during the previous inter-sample interval; this is labelled DELTAX. DELTAX is obtained by sampling the output of position counter **72** at each sample time (i.e., at the beginning and end of that interval). DELTAX is a signed variable and its sign indicates the direction of carriage motion during the interval time. Next, if required by the nature of the printer mechanism, an appropriate offset may be added to DELTAX, in step **320**. The purpose of the offset is to compensate for mechanical "play" in the printer drive or other such factors which are functions primarily of direction and not velocity. In some systems, such offsets may not be needed for acceptable registration of printed dots. The next step, **330**, is to compute the change in FTC required by virtue of the change in the offset-adjusted DELTAX observed during the last sampling period. The required value of FTC may generally be expressed as a function of offset-adjusted DELTAX. In the particular implementation using a printer mechanism similar to that employed in the model LA36 terminal manufactured by Digital Equipment Corporation, Maynard, MA, it has been found that this function may be expressed as a proportionality constant whose value is primarily determined by print wire flight time and secondarily determined by velocity-related mechanism variations, such as elasticity of the carriage driving linkage. Thus, the FTC connection function may be obtained from an expression in the form:

$$\text{desired } FTC = KFTC\,(\text{DELTAX} + \text{OFFSET ADJUSTMENT}),$$

where KFTC represents the proportionality constant.

Since the transition counter can only resolve integer multiples of a transition and the above calculation may lead to the computation of a desired FTC which differs from the position count by a non-integral number of transitions, the desired FTC can only be approached on a greatest integer basis. Specifically, the resolution obtained in the above calculation is equal to one transition divided by KFTC. Thus, in step **340**, an add/subtract command is sent to the T&IC **86** to cause the net transition generator to add or subtract a number of transitions corresponding to the greatest integer in the difference between the prior FTC and the desired FTC, to cause the greatest integer in the desired FTC to become the new FTC.

While it may be possible to complete the above updating of FTC in its entirety during one inter-sample period, it has also been found that if acceleration is low enough, the updating may be done at a slower rate, even as slow as one transition add or subtract per sample interval.

With the foregoing as background, a specific embodiment of circuitry for the printer control circuit will now be described. First, however, to aid in understanding the actual circuitry, certain aspects of the system will be explained in greater detail and the circuitry will be introduced in the form of an apparatus block diagram. With that foundation, the detailed circuitry will be readily explainable and understandable.

FIG. **14** shows a system level block diagram in slightly greater detail than that shown in FIG. **1**. FIG. **14** differs from FIG. **1** principally in that it shows the type of signals to be transmitted over the data bus **58**. The printer control circuit is also shown in greater detail, illustrating its various functional subsections.

As shown in FIG. **14**, the printer control circuit (PCC) **30** has six subsections: a dot print control section **30A**, carriage position count section **30B**, carriage speed control section **30C**, line feed control section **30D**, bell control section **30E**, and frequency generation section **30F**.

The PCC **30** performs functions which are beyond the capabilities of the microprocessor **52** and provides an interface between the microprocessor and the printer electromechanical components.

The dot print control section **30A** provides the print head control signal which appears on line **44**. It receives horizontal pitch information and dot data over the bus and supplies dot interrupts to the microprocessor to request additional dot data and signal when the dot buffer is empty.

The carriage position count section **30B** receives the output signal on line **42** from the incremental shaft position encoder **26B** and supplies a position signal to the bus, indicative of the carriage position.

Carriage speed control section **30C** receives the speed command from the bus and supplies the carriage drive control signal on line **36** to the carriage motor amplifier **38**.

The line feed control section **30D** receives line feed step instructions from the data bus and supplies a line feed control signal on line **32** to line feed amplifier **34**. The line feed control section also supplies a line feed interrupt to the processor, via the data bus.

The bell control section **30E** drives a bell amplifier and loud speaker to provide an audio signal responsive to a bell command.

The frequency generation section **30F** contains an oscillator and countdown chain which provides required clocking signals most of which feed elements internal to the PCC. The frequency generation section also provides tick interrupt signals at 2.5 ms intervals to the data bus. In addition, it supplies baud rate clocks directly to the input interface **48** responsive to baud rate selection signals received from the bus.

Microprocessor **52** receives interrupt requests and interrupt vectors from the printer control circuit (PCC) **30** and input interface **48** via bus **58**. When powering up, the microprocessor reset line is asserted, thus starting program execution from location **0** with interrupts disabled. The PCC **30** will cause the processor to vector to one of four interrupt service routines, depending upon the source of the interrupt request. These are indicated by the table below:

Interrupt Service Routine	PCC Dot Print	Input Interface	PCC 2.5 ms Clock Tick
Tick Service	F	F	T
Input Service	F	T	F
Tick Service	F	T	T
Dot Service	T	F	F
Dot Service	T	F	T
Dot Service	T	T	F
Dot Service	T	T	T

As the above table indicates, the input interface interrupt is masked by either or both of the PCC interrupt sources, and the 2.5 ms clock tick interrupt is masked by the dot print control interrupt.

The PCC **30** is addressed by being mapped into a portion of the available memory address space of the processor. Memory references to addresses 0111 xxxx xxxx xxxx will select the PCC **30**. Address bits **A2–A11** are ignored for both read and write operations; **A0** and **A1** are also ignored for read operations. Due to the slow internal operation of the PCC **30** relative to the speed of bus transactions, write accesses must be at least 7 microseconds apart; there is no similar restriction on read operations. Most write operations to the PCC pass additional sub-addressing data in the data field.

In order to concentrate all of the control functions for the printer into one integrated, circuit, the PCC has been designed to include the following functions, in addition to the dot print control function which is disclosed in detail later in this document: carriage speed control, line feed control, frequency generation, tick alarm (i.e. a processor integrity check), and interrupt vector generation.

Carriage Speed Control

The voltage supplied to the carriage DC motor **26A** is controlled by a pair of pulse streams. The first pulse stream has a duty cycle which is proportional to the binary value of a microprocessor-supplied speed command. The second pulse stream has a duty cycle which is proportional to motor speed. The integrated pulse streams are subtracted and integrated outside the PCC to yield an error signal which is fed to the motor. A steering circuit interchanges the roles of the two pulse streams to provide direction control, as a function of the directions of the speed command and the motor speed.

Line Feed Control

The line feed stepper motor is controlled by a 2-bit state word which controls the polarity of the voltage to the motor windings and the third bit which controls the amplitude of the current in the windings. Higher current is used for running the motor than for holding a steady position.

The timing of state changes during line feeding is completely under control of the microprocessor with the aid of a programmable timer interrupt which is made available by utilizing the countdown timer of the dot print control circuit in a timer mode rather than in a printing mode.

Frequency Generation

Many processes in the printer depend upon timing signals for proper operation. The master source of timing is an 18 MHz crystal which drives the clock. The clock provides a 2 MHz signal which is used by the microprocessor, the input interface, and the printer control circuit as their primary timing inputs. The 2 MHz clock input to the PCC is divided by a factor of 13, yielding approximately 153.6 kHz, to provide the clock which is used to drive most of the sequential logic in the PCC. A divider chain also produces 76.8 kHz, 38.4 kHz, 19.2 kHz, 9.6 kHz, 4.8 kHz, 2.4 kHz, and 1.2 kHz signals. A divide-by-three circuit produces 400 Hz. The signals from 153.6 kHz to 1.2 kHz drive a bit rate multiplier (BRM) in the carriage speed control circuit for the purpose of producing the signal whose duty cycle is proportional to the speed command. The 400 Hz signal provides an interrupt to the microprocessor every 2.5 ms which allows firmware-implemented, microprocessor-controlled processes to run periodically and thus be capable of performing time dependent operations.

A baud rate generator circuit further divides various outputs of the divider chain in order to produce baud rate clock signals for lower baud rates and split baud rates. A baud rate selector circuit allows the microprocessor to select various outputs of either the divider chain or the baud rate generator to be supplied as baud rate clock signals to the input interface.

Tick Alarm

In order to prevent a microprocessor failure from causing physical harm to power circuits or electromechanical components, the PCC has a special monitoring function which requires the microprocessor to prove its integrity by responding to a clear tick command (i.e., interrupt signal) every 2.5 ms. Failure to respond is proof of a malfunction and results in the disabling of most PCC outputs. The PCC powers-up with this so-called tick alarm activated, so that outputs are disabled until a programmed initialization sequence has been completed.

Firmware Overview

Scheduling

Firmware processes run at one of four priority levels, at any instant of time; they are, in descending order: interrupt level (with interrupts disabled), tick level,

interlaced tick level and background level. Except for interrupt service routines, interrupts are normally kept enabled. The hardware interrupt vector scheme subdivides interrupts into two priorities, with input interface interrupts below all others. Firmware then further divides the higher level so that dot or timer interrupts take priority over tick interrupts.

Processes that run at background are the least time-critical, since background may be locked out by higher priority processes for short periods of time. On the whole, though, background gets the lion's share of processor time, typically more than 50%. Tasks that run in background include line image formatting and print scheduling. All printing, slewing and vertical motion is originally scheduled by background level routines even though executed at higher priority. When the processor is idle, it will be looping at background level.

Interlaced tick level processes execute periodically after non-interlaced tick. Every interlaced tick process runs once every one, two, four or eight ticks, where a tick is 2.5 ms. This is accomplished by a round-robbin scheduler, where each of eight phases is executed in turn on successive ticks. Each phase runs to completion before the next phase is begun. Examples of interlaced tick processes are the communications protocol handler for the input interface and the polling of interlock switches and operator controls and indicators.

Non-interlaced tick processing is begun immediately after taking a tick interrupt; interlaced tick follows its completion.

Servo control comprises all of non-interlaced tick processing. This level of processing must be guaranteed to finish before the next tick interrupt comes along.

Interrupt service routines are the most time critical and are kept as short and fast as possible. They include the dot buffer interrupt, timer mode interrupt, and input interface receiver ready interrupt.

Processes like initialization require that interrupts be off while running and thus effectively run at highest priority.

Print Control Firmware

One of the more complicated operations in the printer firmware is the starting of the print operation after a carriage return or tab. The PCC will only begin the printing operation after it has been given a print start command. This command will be executed at the next point at which printing legitimately can be started.

Starting the printing operation

The task of starting the print operation is handled in two steps. The first step is to prepare the PCC to begin printing. This is done once for every print start. On the next and subsequent ticks, tests are run to see whether the carriage has arrived at the proper location to begin printing and that it is travelling at the proper speed. These tests are repeated until printing is begun.

The forward printing process is begun by obtaining the first character to be printed from the line buffer and loading its dots into the dot buffer. The first four sets of dots are then loaded into the fifo of the PCC, the dot interrupt is enabled and the carriage is started on the approach to the starting point of the print string.

The reverse printing process is begun the same as that described above, except that all operations are for backward printing. The right-most character in the line is loaded into the dot buffer.

Print start is enabled by a print start command which is stored by the PCC and printing is begun when the print head wires cross the boundary of the intercharacter space in either direction. To initiate a forward print start properly, the command must be given before the impact point crosses from the intercharacter space of the previous character into the cell of the character to be printed. (The impact point is defined as the point on the paper at which the print wire would strike if its solenoid were energized at the then current position of the print head.)

As a practical matter, the print start command is given only when the impact point is in the Safe Print Start Zone and the carriage is moving in the proper direction. The Safe Print Start Zone begins three transitions to the right of the left hand cell boundary of the previous character and ends three transitions to the left of the intercharacter space of the same character cell.

This scheme assures that the print start is only executed when the impact point crosses the left hand cell boundary of the character to be printed. Premature print starts are thus avoided, as are print starts in the reverse direction which might otherwise be induced by unintended motions in the reverse direction. Such undesirable print starts would cause the characters printed subsequently to be displaced. Tests are therefore made by the microprocessor to assure that the carriage is travelling to the right at a speed which does not exceed the print speed by more than a small amount, that the impact point is in the Safe Print Start Zone, and that other conditions outlined below are met.

Other tests must be made before forward printing may actually be started. The first test is to see whether a vertical motion is in progress or pending, a cover is opened, paper is out, etc. If any of these conditions are true, the routine is aborted. Next the direction in which the carriage is travelling is checked. If it is negative, the routine is aborted, otherwise the tests are continued by computing the maximum permissible approach speed as a function of the print speed limit currently in effect. Then the position counter is read and a projected impact point is computed. If the impact point is to the right of the Safe Print Start Zone, the carriage has overshot the mark and the routine is aborted. If the impact point is more than 255 transitions to the left of the right hand border of the Safe Print Start Zone, the routine exits and no further tests are made during this tick interval, since there is time to do the tests during the next tick interval. If the impact point is within range, the following tests are made:

1. a check that the carriage speed is less than the maximum allowed;

2. a determination that the impact point is less than 34 transitions to the left of the right hand border of the Safe Print Start Zone; and

3. a test that the print head is not likely to stop or change direction in the near future.

If any of the above tests are failed, printing cannot be started at the current tick time, and the routine exits; otherwise, a go ahead for printing is issued and a final test is made to assure that the impact point is to the right of the left hand boundary of the Safe Print Start Zone. In the event that this test is failed, the routine aborts. If the test is passed, the print start command is given to the PCC and the address of the dot service routine is set up in the dot interrupt service routine and the printing operaiton is finally begun.

The procedure for initiating a reverse print start is the same as that used for the forward print start except that

direction of motion is from right to left instead of from left to right.

Carriage Servo Firmware

The carriage servo firmware ascertains the current position of the carriage and generates the speed commands needed by the carriage servo to move the carriage to the position where it should be. Actual and desired positions are determined every time a tick interrupt is received, which is every 2.5 ms. The speed command given to the servo is then updated on the basis of these position determinations. This command is such that the carriage servo moves the carriage toward the desired position as rapidly as possible. As the carriage approaches the desired position, the speed is reduced to allow for a smooth stop.

The position of the carriage is measured by an incremental two-channel encoder mounted on the back of the carriage servo motor. Each channel has a square-wave output which produces 660 cycles per revolution, in phase quadrature. A transition is defined as the change of the output of one channel from the logical zero level to the logical one level, or vice versa. Thus, there are two transitions per cycle in each channel and since there are two channels, there are a total of four transitions per cycle and 2640 transitions per revolution.

The motor moves the carriage 4 inches per revolution. Hence, there are 660 transitions per inch of carriage motion and one transition equals 0.001515 inches. The carriage servo routines are entered every time a tick interrupt is received. The tick interrupt is cleared and the actual position of the carriage is determined. Speed, flight time correction, carriage position, and error are determined. A real time clock is also incremented.

Upon entry, the position counter in the PCC chip is read. Interrupts are then enabled. The previous reading of the position counter is retrieved and subtracted from the current reading of the counter and the difference stored for use in subsequent calculations. Since readings are made at fixed time intervals, the difference is a measure of the carriage speed. The value of the current counter reading is stored for use during the next tick interrupt service. The calculation is valid as long as the carriage did not move more than 127 transitions during the preceeding tick interval.

The determination of the command speed for the carriage servo is begun on the basis of how far the carriage is from where it is supposed to be. The actual target speed for the servo is obtained from a look-up table. This target speed is the speed input for a subsequent routine, where it is processed to generate the actual servo command. The sign of the target speed is chosen so that the carriage moves toward the point at which it is supposed to be and the target speed decreases as the carriage approaches its destination, so that it arrives there at a reasonable speed.

In general, the speed command is based on the target speed unless the change from the previous target speed was too large, in which case an acceleration limit is applied.

A speed error is computed by comparing measured carriage speed with the target speed. If this error exceeds the limit, it is an indication that the carriage is not moving as fast as it should be and that it may in fact be stopped completely. If the error exceeds the limit for a predetermined number of ticks in succession, it is presumed that the carriage is stuck. This causes a pause flag to be set; other pause flags are set when there is no paper or when the cover is opened. Before a speed command is sent to the PCC, the pause flags are examined. If no pause flag is set, the speed command determined as previously described is sent. If any pause flag is set, a speed command of zero is sent.

Dot Rate Limiting

As described above, there is a possibility that the print head could overheat if called upon to print excessively dot-dense text. Provision is therefore made in the firmware to monitor the rate at which dots are printed. If the rate exceeds a predetermined limit, the print speed is lowered. A suitable integration is performed to prevent short bursts of heavy printing from causing an unnecessary reduction in speed.

Character Processing

When a character is received and fully assembled by the input interface, an interrupt is sent to the microprocessor. This interrupt is serviced by the character reception routine. This routine reads the character from the input interface and checks for parity errors. If a parity error is detected, the character is converted to a SUB character. If the character is neither NUL or DEL (the all zeros and all ones codes, respectively) it is stored in the input buffer in RAM to be processed by the background executive routine.

Characters are processed by the background executive routine. The source of characters may be either the input buffer or (optionally) the local character slot, which may contain characters generated by operator controls. Local form feed and local line feed keys (typically found on terminal-type-printers) use the local character slot to merge their requests with characters from the input interface.

When the background executive has a character to process, depending on the mode and state of the printer, it forwards it to either the print line builder, escape sequence or the command processor, which interprets escape sequence commands which forwards it to either the print live builder or the escape sequence processor, which interprets escape sequence commands. Escape sequence commands are the user's means of specifying parameters such as character pitch, margins, tab stops, line spacing, and page length.

Principal characters are routed to the print line builder. Upon receiving a character, the first task performed by the print line builder is to determine if it is a control character or a graphic character. If the character is a control character, it is used to index a table of control routine addresses and an appropriate control routine is executed. For example, routines are required for handling horizontal and vertical tabs. They call other sub-routines which search a tab bit map table to find the first tab set past the current printing position. Each routine returns with a value that specifies what the active line or columns would be when the routine has finished. If the character is a graphic character, it is moved into one of two print line buffers after a certain amount of processing.

The processing of a graphic character requires look-up tables to convert certain codes in the various national languages to an internal code which will point to dot matrix representation of the character. Also, if the printer is in the alternate character set, the character is converted so that it points to the alternate character set dot table.

The print line buffer is then checked to be sure that the active column (i.e., the column in which the character is supposed to be deposited) is within the left and right margins. If it is outside these margins and auto-new-line is enabled, the character is not deposited in the current print line. The current print line is forced to be printed followed by a line feed. The active column is returned to the left margin. The result is as though a carriage return and line feed sequence was received just before the character. The character is then put into the next print line buffer at the left margin. If the character is outside the margin and the auto-new-line is disabled, the presence of the character means that printing is being attempted beyond the right margin. The character is discarded and an alarm is sounded.

Before attempting to deposit a character in the print line buffer, a test is made to be sure that there isn't a character already in the buffer at the active column. If there is, an overprint condition exists. Detection of an overprint prevents adding any more characters to the print line buffer until it has been emptied. A new print line buffer is made available and the character is written into the new buffer at the active column. Depositing a character in the print line buffer causes the active column to be incremented by one. Tests are performed to determine if the right margin warning alarm should be sounded and to see if the right margin has been reached. Other types of information stored in the print line buffers have to do with producing vertical motion between lines and sounding an alarm. Thus, a complete print line contains three types of information: alarm count, line feed step count and character codes. The character codes are positioned in the print line buffer at the same spot that they would appear on the paper. When a print line buffer is full or its printing is forced (due to overprinting, etc.), then the print line buffer is given up by the print line builder and traded for an empty print line buffer. When the full print line buffer is acted on by the printing routines, it has to be prepared before it can be used to start printing characters.

When a line is to be printed, a decision has to be made as to which direction the line is to be printed. A line is always printed forward unless it has vertical motion both before and after it. Also, a line is never printed in reverse which contains overprinting (i.e., a slash over a 0). Recall that a print line buffer is forced to print when overprinting occurs. When forced to print, the buffer is swapped with the empty buffer and the printing routines take over. Since the forced-to-print-buffer is not full, it doesn't have a vertical motion code at the end of the buffer. It has been sent to the printer for overprinting, not because there is a terminating line feed character at the end of the line. Consequently, the direction decision algorithm would find that this print line buffer does not have vertical motion both before and after it.

If the print line buffer is a complete line bracketed by vertical motion on both sides, then the print direction is decided by the direction algorithm. The algorithm remembers the last column printed on the previous line. If the previous line had characters from columns 1 to 100 and was printed from left to right, then the last column was 100. If the next line has columns from 25 to 125, the direction algorithm finds which of the limits is closest to the last column of the previous line. In this example, 100 is closer to 125 than 25, therefore the new print line is printed from right to left.

After the print direction has been established, it is necessary to look at the print line buffer in the interval between left-most and right-most characters to determine if a long sequence of empty positions is in the print line. The interrupt routine could search through as many as 215 empty positions to locate the next printable character. To prevent this, the firmware performs a housekeeping routine which scans the print line and converts the empty positions to real spaces (octal code 40) or inserts a special tabing command. The tabbing commands are interpreted by the printing routine to indicate that there is a long sequence of white space, to stop printing and start slewing the printhead. The routine also specifies where to stop slewing and again start printing characters. As an example, let us suppose a print line has characters from columns 1–10, then blanks from columns 11–100, and characters again from columns 101–110. The housekeeping routine would search the columns until it got to the blank in column 11. It would then start to count blanks. If it finds five empty positions in a row, the routine leaves the printing mode and executes a high speed slew to the next column in which characters will be found. When it finds the next character (in column 101), it will go back to column 11 and write a -1 value in that slot. This flags the column as the beginning of a high speed slew. During printing, when the print routine encounters the -1, it causes the print routine to stop and pass control to a high speed slew routine. The next column (12) contains the address of the column (101) in which is located the next printable character. Thus, the "look-ahead" housekeeping routine functions to speed up the rate or printing.

When the print line buffer is completely filled and the printing interrupt routine is free, the filled print line buffer is passed over the background printing routine. If there are any alarm signals specified in the print line, the code is passed over to the alarm or bell handling routine to sound a bell at a 400 to 2400 Hz rate at the proper time. If there are any line feed steps to be performed, the line feed routine will be called, at the appropriate time, to index the stepper motor. Any characters contained with the print line will call the printing routines which will be scheduled at the appropriate column and printing will occur. When the print line buffer has been completely processed by the printing routines, an executive routine pointer will exchange the empty print line buffer for the buffer just filled. The empty buffer will be zeroed and returned to the background where it will start accepting new characters.

Having thus described the operation of the printer and the PCC, in particular, and the relationship between the processor/firmware and the PCC, a detailed explanation of the PCC hardware follows.

Attention is first directed to FIG. **15**. FIG. **15** is a slightly modified version of FIG. **4** which more explicitly shows the interconnection paths between the increment steering circuit **104** and the dot buffers and timers **120A** and **120B**. Both FIG. **4** and FIG. **15** are functional block diagrams which are highly useful in explaining the operation of the PCC. The circuit structural elements of the PCC, however, do not correspond exactly, block-for-block, to the functional assemblies illustrated in those figures, since the functional block diagrams were dervied from the actual circuitry in order to facilitate understanding of PCC operation. A corresponding apparatus block diagram for a suitable hardware implementations of the PCC is provided in FIGS. **16A–16D**. While all of the functionally of FIGS. **4** and **15** is provided by the apparatus of FIGS. **16A–16D**, the apparatus of the latter figures further provides additional func-

tionality which is not shown in FIGS. **4** and **15** and which is not necessary for the proper operation of the printer (e.g., the tick alarm feature, bell generation, etc.).

Reference is now made to FIG. **16A**. In that Figure, a one block encoder signal filter **204** is shown, corresponding to the two separate encoder signal filters **204A** and **204B** of prior Figures. Sampling signals for the encoder signal filter are provided on lines **502–506**. The outputs from the encoder signal filter **204** are supplied via lines **214** and **216** to the transition detector **508**. The latter corresponds to the transition detectors **206A** and **206B** and the direction decoder **202**. Thus, the outputs of the transition detector **508** are provided on the same lines **82**, **84** and **92** which are shown as the outputs in FIG. **7**.

Clock generator **510** receives as an input on line **512** a 2 MHz nominal square wave signal designated CLK. From this input, the clock generator provides non-overlapping 2 MHz clock signals, PHY and PHX which appear, respectively on lines **514** and **516**. The PHX and PHY signals comprise the internal 2 MHz clock signals used by the various elements of the PCC. The rising edge of CLK generates the falling edge of PHY. The falling edge of PHY generates the rising edge of PHX. The falling edge of CLK generates the falling edge of PHX. The falling edge of PHX generates the rising edge of PHY.

In the clock generator, a pair of capacitors are slowly charged toward a high level by the PCC supply voltage, Vcc, and are rapidly discharged to a low level via PHX and PHY, respectively. If either capacitor ever charges to a high level, then a high level is presented as an output signal, designated CLK FAIL, on line **518**, to indicate a failure of PHX or PHY clock signals. So long as the CLK input signal continues to oscillate at or near a 2 MHz rate and the PHX and PHY signals are generated, the CLK FAIL output signal will remain at a low level.

Four additional clocking signals, designated **PH1**, **PH2**, **PH3** and **PH4** are also generated on lines **522**, **524**, **504** and **506**, respectively. The signals PH1–PH4 are generated by a four-bit shift register whose states can be represented by the numbers 0 through 15 and whose major state transition sequence is 0-14-7-11-13-6-3-9-4-10-5-3-1-0. Minor sequences of 8-12-14 and Reset-15-7 both enter the major sequence. State transitions occur every 500 ns, thus providing a major sequence repetition period of 6.5 ms. The states are decoded to provide the PH1–PH4 clock signals at a rate of approximately 153.6 Khz. State 3 provides **PH3**, state 9 provides **PH4**, and state 0 provides **PH2**. **PH1** is provided by a one-bit delay of **PH2**. A further output signal, PH1 DUE is provided on line **532** to provide an indication that a **PH1** pulse is about to occur; this signal is used to reset the write control circuit **610**.

The reset circuit **540** samples a RESET input signal on line **542** at the 2 MHz clock rate of the PHX and PHY signals. It provides an EARLY RESET signal to the clock generator **510** and encoder signal filter **204** via line **502** when a reset triggering signal is impressed on line **542**. In addition, the early RESET signal causes the reset signal on line **544** to be on, to reset counters and logic throughout the PCC. When the early reset signal is off, it allows the clock generator to begin issuing **PH1** and **PH2** clock pulses. The second occurence of the **PH1** clock pulse causes the RESET signal on line **544** to be turned off. This event terminates the reset state at all circuits that use the reset signal and allows normal operation to commence.

The interrupt control circuit **550** is the gateway through which interrupt request signals leave the PCC. Two type of interrupt request signals are generated in the PCC, tick interrupt requests and dot buffer/timer interrupt requests, designated respectively TINT REQ and DINT REQ. The TINT REQ signal is provided to the interrupt control circuit on line **552** and the DINT REQ signal is provided on line **554**. The interrupt control circuit **550** samples these two interrupt request signals and the logical OR of the samples is presented on the interrupt (INT) output line, **556**. The sampling occurs at the PHX–PHY clock rate, except during the interrupt acknowledge (INTA) pulse received in inverted state on line **558**. The samples are held steady during the INTA pulse to provide steady interrupt vectors on lines **562** and **564** during the interrupt acknowledge cycle. Line **562** provides the interrupt vector from the dot buffer and timer upon request, designated D/T INT, and line **564** conveys the tick interrupt vector, designated TICK INT. These interrupt vectors are supplied to the data output multiplexer (MUX) **560**, through which they are placed on the data bus to be read by the processor.

Data output multiplexer **560** comprises an 8-bit, 2-to-1 multiplexer which propagates the read data from read data latch **570** to the data output buffer **580** except during the INTA pulse, when it propagates the interrupt vector instead. For this purpose, the data output multiplexer receives the −INTA signal on line **558** and the D/T INT and TICK INT signals on lines **562** and **564**. The interrupt vector consists of all one's except bits **4** and **5**, which represent the TICK and DOT/TIMER interrupts, respectively.

The data output buffer **78** is a buffer comprised of 3-state drivers which present the position counter reading (through read data latch **570**) or the interrupt vector onto the data output lines D0–D7 of the PCC, when enabled by an output enable signal on line **582** from the read control circuit **590**.

Read control circuit **590** provides the output enable signal on line **582** which allows the data output buffer **580** to drive the data output lines D0–D7 during read and interrupt acknowledge cycles. It also provides a HOLD READ DATA signal on line **572** which strobes the read data latch **570** and thereby captures the position indicated by the position counter and prevents the position counter reading in the read data latch from changing during a read cycle or at any time that a tick interrupt request is pending. Holding the position count for reading purposes while a tick interrupt is pending makes the firmware speed calculations independent of interrupt service latency time.

As implied above, the read data latch **570** is simply a buffer for reading in parallel the position counter reading. The read data latch continuously keeps track of the position counter reading except during the on condition of the HOLD READ DATA signal, during which time its reading is frozen.

The position counter **72** is an 8-bit binary up/down counter which is set to all one's by the reset signal on line **544**. The signal on line **84** provides the events that are counted. The signal on line **82** provides the direction or sign of the event.

Write control circuit **610** samples the signal on line **612**, designated −WR, at the clock rate of the PHX and PHY signals, responsive to the PH1 DUE signal on line

532. The event of writing a command is detected and held until the next cycle at the PH1, PH2 clock rate. The write signal causes the command decoder 66 to issue a pulse on one of its 21 output lines, thus writing the data into one of the circuits. A write edge signal is provided to cause the write data buffer 620 to latch the address and data inputs for subsequent use during the assertion on the write signal or line 616.

Write data buffer 620 monitors the address lines A0, A1 and data lines D0–D7 and latches them during the occurrence of the write edge pulse on line 614. The output of the write data buffer is fed onto the internal PCC bus 64.

Command decoder 66, as its name implies, decodes the signals on lines D0–D7, A0–A1, and provides an output pulse on one of 21 lines during the write pulse on line 616, as a function of the address and data values indicated on the D0–D7, A0–A1 lines. Lines D0–D7 and A0–A1 provide the PCC connections onto the printer bus 58.

Reference is now made to FIG. 16B. The transition detector 508 from FIG. 16A supplies its output signals on lines 82 and 84 as inputs to transition and circuit 720. Transition add circuit 720 corresponds substantially, but not exactly to transition adder 252 of FIG. 11. The discrepency is principally that the net transition generator 252A is separately shown in FIG. 16B whereas it was included as part of transition adder 252 in FIG. 11.

The transition add circuit 720 is used to change the state of the transition counter 254 under firmware control. This change is effected by causing the transition count in the transition counter to be adjusted by the addition or subtraction of a desired number of transitions.

The transition add circuit 720 (see FIG. 18Z) contains a latch and a 4-bit synchronous counter. The controlling command, W ADD, is provided on line 722 from the command decoder 66. The command is in the sign-magnitude format, with the magnitude being negated in two's complement form. When a command is strobed in by W ADD, ADD PENDING and the transition signal which appears as an output on lines 724 are asserted, causing one transition to be counted by the transition counter for each cycle of the PH1 clock. If there is no transition on line 84 at this time, then output line 726 is set to the states corresponding to the command. The counter is incremented once per cycle of PH1 until a count of zero is reached, at which time the internal ADD PENDING signal is negated, thereby causing the transition signal on line 724 and the internal ADD-IN-PROCESS signal to be negated. The counter then stops.

It can be seen that in the mode of operation described above, the transition counter and the counter in the transition add circuit both change by one count when PH1 is asserted, as long as the count of the counter in the adder is not zero. In this manner, the comand is transferred to the transition counter 254.

If there should be an encoder transition during the process described above, the ADD-IN-PROCESS signal is negated and counting by the up counter is suspended. Line 726 is set to the state of line 82 and the encoder transition is counted. When there are no more encoder transitions, the transition add process is resumed, as described above.

Net transition generator 252A samples the TRANSITION RIGHT signal on line 726 whenever a transition occurs. The sample forms NET RIGHT and NET LEFT signals which are provided on line 253.

A 1-bit toggle to allowed to toggle each time the direction of a transition differs from the stored direction of the previous transition. The output of the toggle forms a REVERSAL PENDING signal. (The initial value of the REVERSAL PENDING signal is false.) Any transition which causes the REVERSAL PENDING signal to toggle is not allowed to propagate to the NET TRANSITION signal on line 98. Any transition which occurs while the REVERSAL PENDING signal is true (and which was allowed to propagate to the NET TRANSITION signal) causes the NET TRANSITION signal to be asserted for an additional clock period, thus propagating the transition that was stored in the reversal pending memory. The propagation of this transition clears the reversal pending memory.

Transition counter 254 receives preload values for assigning values to the parameters TPI and TPS from bus 64 responsive to the W TPI and W TPS signals on lines 727 and 728, respectively. The latter two command signals are provided under firmware control, over the bus 58, and are detected and decoded by the command decoder 66. The increment couner 256 provides a signal, SEL ICS, to the transition counter 254 on line 732. The SEL ICS signal controls a 4-bit multiplexer in the transition counter for presenting either TPI or TPS to that counter as radix information. The transition counter includes a 4-bit up/down counter which contains an exclusive-OR function in each stage. This exclusive-OR function compares the counter bit value with the radix bit value for that stage. When all four such states detect a matched condition, an internal signal designated T=MAX is true. When all four counter bits are zero, a signal T=0 is true. The control inputs of the counter are a toggle command, an up/down indication and reset and load commands. The NET RIGHT signal on line 253 provides the up/down count control indication. According to the convention adopted. counting up corresponds to rightward motion of the print head.

A net transition is counted by causing a toggle command, unless the T=MAX signal is true and the count direction is up, in which case the counter is reset to zero; if the T=0 signal is true and the count direction is down, the radix data from the multiplexer is loaded into the counter.

Either of the above exception events, (i.e., load or reset) constitutes a primary increment and is propagated to the PRIMARY INC signal on line 102 for use by the increment counter 256, the print enable function of mode control circuit 740, the increment steering circuit 750 and the increment shift register 100, the later two blocks being shown on FIG. 16C.

The LD T and I signal provided by the command decoder on line 734 allows the firmware to initialize the transition counter to any desired value. (Note that when the state of the signal on line 732 is not known, the TPI and TPS values must be identical for this operation.)

Increment counter 256 corresponds to that previously shown in FIG. 11. The desired increment counter preload value to set the radix of the counter equal to the parameter IPC is loaded under firmware control. The command signal is given through the command decoder on line 736 and is designated W IPC. The actual radix value is transmitted to the increment counter via bus 64. The increment counter comprises a 5-bit up/down counter which contains an exclusive-OR function in

each stage. The exclusive-OR function compares the counter bit value with the radix bit value for that stage. When all five gates detect a match, the I=MAX signal, provided as an output on line **738**, is true. When all five counter bits are 0, the I=0 signal is true. The control inputs of the counter are a toggle command, an up/down indication, and reset and load commands. The net right signal on line **253** provides the up/down indication, with counting up corresponding to rightward motion of the print head. The print enable function in the mode control circuit **740** receives a character boundary indication whenever a primary increment occurs with the I=MAX signal being true. Thus, printing begins when exiting the intercharacter space in either the forward or reverse direction.

An increment is counted by causing a toggle command, unless the I=MAX signal is true and the count direction is up, in which case the counter is reset to 0, or unless the I=0 signal is true and the count direction is down, in which case the IPC value is loaded into the counter.

Either of the above exception events (i.e., load or reset) constitutes an intercharacter space and is propagated to the SEL ICS signal for use by the transition counter.

The LD T & I signal allows the firmware to initialize the increment counter to any desired value.

Departure from the functional block diagram of FIG. **4** has heretofore been minor. In the following discussion, however, more significant departures will be made in presenting the circuitry which performs the functions implemented by the blocks in FIG. **4**.

By way of introduction to the explanation which follows, it will be observed that the dot buffer and timer blocks **120A** and **120B** are broken down into separate dot buffers and timers and that the control signals for those elements are shown in greater detail. Also, there is no print enable circuit standing alone, as the enabling of printing is a function performed by block **740**, the mode control circuit, which includes within it a print enable circuit and elements of the increment steering function of block **104**.

Timers **760** and **770** can be operated in different modes, responsive to signals provided by mode control circuit **740**. Five modes are available, but only the three modes which constitute active print modes are necessary to the present explanation. The functionality provided by the other modes is not necessary to the printer operation and will be described out in passing. Mode control circuit **740** performs the print enable function, as above noted, as well as providing mode selection for both the primary/secondary increment steering control **750** and timers **760** and **770**.

The print enable function has a single latched output, the DOTX signal, provided on line **742** to timer A controller **762** and timer B controller **772**. The DOTX signal is off when printing is not enabled. When one of three printing modes is selected, the first primary increment that appears while the increment counter is at its maximum value will cause DOTX to turn on. DOTX remains on until either a fifo empty indication occurs or the printing mode is cancelled. The FIFO EMPTY signal is received by mode control circuit **740** on line **106**, while printing mode initiation and cancellation signals are received on line **746** as the state of the W PT ST signal from the command decoder.

Operation of the primary/secondary increment steering control is governed by of a 2-bit start command (W PT ST) which provides the data for the PRIM **6** and PRIM **5** output signals on lines **748** and **749**, respectively. The PRIM **6** and PRIM **5** signals inform the primary/secondary increment steering control **750** as to which of the two columns of the print head (in a two column embodiment) is the leading column, so that it can direct the primary and secondary increment signals to the appropriate columns.

Mode selection is defined by the combination of states of the PRIM **5** and PRIM **6** signals. If either of these signals is on, a printing mode is selected. A print start command of 00 (i.e., both PRIM **5** and PRIM **6** in a 0 state) is interpreted as a print stop signal and is additionally decoded to provide a signal, CLR FIFO, on line **752**, to initialize the dot buffers **810** and **820**.

Mode selection for the timers is provided by the printing mode selection circuit **740** and an additional 1-bit command, W TIM MODE, which is used to initiate a general purpose timer mode of operation for the timers. It is an undefined condition for both the timer mode and a printing mode to be on at the same time. If both the timer mode and all of the printing modes are off, a NO MODE signal is presented on line **754**, to reset the timers to the done state when they are not in use.

The operation of timer A, **760**, is further controlled by timer A controller, **762**, and preload latch A, **764**. Correspondingly, timer B, **770**, is also controlled by timer B, controller **772** and preload latch B, **774**. Preload latch A, **764**, is used mainly to hold a preload count for timer A, **760**. However, it is also used to preload the six low-order bits of the increment shift register **780** when required. The preload latch **764** is loaded from the internal PCC bus **64** in the usual and conventional manner. The outputs of preload latch **764** go to timer **760** and to the low-order six bits of increment shift register **780**.

Timer **760** is a 6-bit preloadable up counter used to generate a predetermined time interval by counting cycles of the 76.8 kHz clock signal F6, appearing on line **766**. The load input of timer **760** is controlled by the signals supplied by timer A controller **762** on line **768**. When the signal on line **768** is asserted, the counter is loaded from preload latch **764** with the desired number of cycles to be counted, in two's complement form. The contents of the counter are sampled by an OR circuit which asserts a signal, RUN **1**, on line **782** as long as any stage of the counter is not 0. The RUN **1** signal gates the F6 signal, causing the counter to run until the count reaches 0, signifying that the preloaded number of cycles have been counted and that the desired time has, therefore, elapsed. At that time, a pulse is placed on line **784** which provides a signal, designated CARRY **1**, to indicate the end of the interval. The NO MODE signal resets the counter to the all 0 state.

The basic function of timer A controller **762** is performed by circuitry which accepts an enable, a run status, and a start command as input, and delivers a load command on line **768** as output. The additional logic of timer A controller allows timer A to work either in printing mode or timer mode, as explained earlier. In printing mode, the INC **6** signal on line **786** starts the timer. The INC **6** signal is supplied by the primary/secondary increment steering control **750**. The starting of the timer consists of waiting until the timer is not running (which requires a 1-bit memory), then issuing a pulse on the LOAD **1** signal on line **768**; this pulse loads the output of the preload latch **764** into the timer **760**, which is actually only an counter. The carry output of

the high order bit of the timer, supplied on line **784** as the CARRY **1** signal, propagates through the timer A controller **762** to become the DONE PULSE **1** signal on line **788**.

There is a provision for overrunning start requests which can occur if the instantaneous print speed is too high. If a second start request occurs with the timer running, a load occurs immediately and a DONE PULSE is produced, thus aborting a dot enable cycle early, but never missing a complete cycle due to the overrun.

In timer mode, the special circuitry which delivers the load command is disabled and timer A runs any time that timer B, **770**, is running. Each time that timer A stops running, if timer B is still running, a load pulse is produced, thus causing timer A to run again. Also, a pulse is produced on the COUNT **2** signal on line **792**, which is supplied to timer B to cause it to count.

Timer B is similar to timer A except that it is also used as the high order part of a 12-bit counter used for timing long time intervals for line-feed control. The signal to be counted and the gating function for this counter are therefore generated outside timer B. The preload is obtained from preload latch B, **774**. The outputs of timer B are a CARRY **2** signal supplied on line **794** to timer B controller **772** and a RUN **2** signal provided on lines **796** to timer B, timer No. A and dot buffer control circuit **800**. The CARRY **2** signal propagates through timer B controller **772** to become the DONE PULSE **2** signal supplied to dot buffer control circuit **800** on line **798**.

The timer B controller **772** contains separate circuitry for timer mode operation and printing mode operation. In printing mode, the circuitry for the timer mode is disabled and operation is similar to the operation of the timer A controller. The INC **5** signal on line **802** provides a start request which is combined with the RUN and CARRY inputs to yield load and done pulse outputs.

In timer mode, a command signal, W PL B, received on line **804**, provides the start request, one cycle delayed from the loading of the preload latch **774**, so that the data will be available to be loaded into the timer. The TIMER MODE LOAD signal provided by timer B controller **772** to timer A controller **762** on line **806** allows timer A to get loaded each time timer B gets loaded. Timer B controller **772** also provides a signal designated TIM BUF EMPTY on line **808** to dot interrupt control circuit **810**, to request an interrupt, which is the ultimate output of the timers in timer mode.

Preload latch B, **774**, is functionally the same as preload latch A, **764**, except that it is used to preload timer B and the high order six bits of the increment shift register **780**.

Increment shift register **780** is composed of dual input flip-flops configured into a 12-bit shift register (only 10 bits of which are actually used) with both serial and parallel input. This shift register is what makes it possible to print dot matrix characters with a print head having two vertical columns of print wires displaced from each other in the direction of travel. Primary increment information is feed serially into the shift register and shifted to the right every time there is a net transition and the printer is in the print mode. The primary increment signal which appears at the output after ten shifts is termed the secondary increment signal and is used to activate the trailing column of print wires. This shifting of information goes on even when a single column head is used, in which case the secondary increment information is not utilized.

When a font is selected, the shift register is preloaded with the increment pattern of that font, allowing secondary increment information to flow to the trailing wires as soon as printing has begun. The length of the shift register can be varied to accommodate different spaces between columns of print wires.

Primary/secondary increment steering control **750** consists of two simple AND/OR gates which direct the primary and secondary increments to the proper dot buffer. If the PRIM **6** signal on line **748** is asserted and the PRIM **5** signal on line **749** is not, the primary increment will appear on line **786** which carries the INC **6** signal, which will send it to dot buffer A **810**, while the secondary increment will go to the dot buffer B **820**. If the PRIM **5** signal on line **749** is asserted and the PRIM **6** on line **748**, is not, the reverse distribution of increments takes place. If both are asserted, both dot buffers receive the primary increment.

Dot buffer controller **800** prevents timer run indications and Done Pulses from propagating to the dot buffers when the timers are operating in the timer mode. It also turns off the dot enable signals when a tick alarm occurs. Dot buffer A, contains a four-level, 6-bit wide, fifo buffer. Data enter the buffer during the W DOT **6** command which appears on line **812**. The DONE **6** signal supplied by the dot buffer controller **800** on line **814** signifies that the output of the buffer **810** is no longer needed, and causes the data in each level of the buffer to advance to the next level. The presence of data in a particular buffer level is indicated by the Q-output of an internal fifo controller associated with that level. These Q-outputs are selectively referred to as DB1 [1:4] and individually as DB11–DB14; they correspond to the signals on lines **122A** in FIG. **4**. These signals are used by other circuits to monitor the activity in the buffer. The CLR FIFO signal on line **752** clears the Q-outputs of the fifo controllers, thus forcing a buffer empty indication on the DB1 [1:4] lines.

Dot buffer B, contains a four-level, 5-bit wide fifo buffer. Data enters this buffer during the W DOT **5** line command signal on line **822**. The DONE **5** signal on line **824** signifies that the output of the buffer is no longer needed and causes the data on each level of the buffer to advance to the next level. The presence of data in a particular buffer level is indicated by the Q-output of an internal fifo controller associated with that level, as in dot buffer **810**. The Q-outputs are collectively referred to as DB **2** [1:4] and individually as DB21–DB **24**, corresponding to line **122B** in FIG. **4**. They are used by other circuits to monitor the activity in the buffer.

The CLEAR FIFO signal clears the Q-outputs of the fifo controllers, thus forcing a buffer empty indication on the Db2 [1:4] lines.

Fifo empty gate **830** consists of a 4-input AND circuit which samples the state of the two blocks or columns in each fifo which are nearest to the output. If none of these blocks contains any information, the FIFO EMPTY signal is asserted on line **744**. The FIFO EMPTY signal is sent to the mode control circuit, which then issues a command to stop printing. Dot interrupt control circuit **815** issues interrupts for both the timer mode and the print mode. The dot interrupt is enabled by setting a latch to 0 from the data bus and is disabled by setting the latch to 1. In timer mode, an interrupt is issued when the TIM BUF EMPTY signal is asserted on line **808**, indicating that an input is needed

for the timers. In dot mode, the interrupt is issued when Db11, Db12, Db21, and Db22 are all asserted, indicating that there is room in the dot buffers for two complete columns of data. The interrupt is cleared by loading data into either the timer buffers (preload latches) or dot buffers, as appropriate.

Dot output control circuit **840** allows the data in dot buffer **810** to propagate to the output pins of the PCC when commanded to do so by the ascertion of a DOT ENA 6 signal on line **842** from the dot buffer controller **800**. Sampling transistors are used between the AND circuit and output devices to eliminate transients in the output when the input data is switched.

Dot output control circuit **850** is similar to dot output control circuit **840** except that it controls only five outputs. Its input is obtained from dot buffer **820** and is controlled by the DOT ENA 5 signal on line **852**.

Carriage motor control is effectuated through speed steering circuit **142** by the provision of control signals on lines **144** and **146**. Speed steering circuit **142** accepts command signals from bit rate multiplier (BRM) **132** and speed feedback signals from pulse stretcher **88**, and routes them to the proper output leads. If the command sign signal on line **134** from the BRM is high, signifying that the motor is to turn in a positive direction, the command pulses on line **136** are gated to the plus lead, **144**, to cause the motor to turn in the positive direction. If the feedback sign signal on line **138** is high, it signifies that the motor and encoder are turning in the positive direction. The feedback pulses carried in the feedback magnitude signal on line **139** are therefor routed to the minus lead **146** so that a signal opposing the motor rotation is generated.

If the TICK ALARM signal on line **872** (from tick interrupt control circuit **880**) is asserted, all of the outputs of the speed steering circuit are disabled.

BRM **132** receives data over the internal PCC bus **64** and clocking signals over PCC clock bus **874**. The signals F [0:6] on bus **874** are square waves obtained from the clock divider **876**. Their frequency decreases in binary progression, with F7 having a frequency of 76.8 kHz, F6 having a frequency of 38.4 kHz, etc. F0 has a frequency of 1200 Hz. The input to the BRM is the speed command transmitted on bus **64**, i.e., bus data lines D [0:7]. The command is in the sign-magnitude format, with D7 carrying the sign and the remaining bits carrying the absolute value of the speed.

Upon receipt of a write speed (W SPD) pulse on line **878** from the command decoder, the data on bus **64** is stored in latches within the BRM. The sign bit on bus line D7 is sent immediately to the speed steering circuit. Nand circuits combine the remaining digits with the square wave signals on the frequency bus **874** to produce the BRM output pulse train. Thus, if bit D6 is asserted, a pulse is injected every time F6 is high. Since F6 is a 76.8 kHz square wave, there will be 76,800 pulses per second, each being 6.5 microseconds wide.

If F5 is asserted, there will be a pulse when F5 is high and F6 is low, forming a train of pulses 6.5 microseconds wide with a pulse repetition frequency of 38.4 kHz. The remaining data bits are treated in a similar maner. Thus, if D0 is asserted, there will be an output only if F0 is positive and F [1:6] are all negative. The net result is a train of 6.5 microsecond pulses with a pulse repitition frequency equal to the frequency of F0. Since none of the pulses generated by the clock divide overlap, the average duty cycle of the signal supplied to center the speed steering circuit by the BRM is directly proportional to the speed magnitude information contained on the lines D [0:6] of data bus **64**.

Pulse stretcher **88** produces a 52 microsecond pulse for every speed transition received, except for the first transition following areveral in direction, which reversal is signalled by a change in state of the signal on line **82**.

The operation of the pulse stretcher is otherwise as indicated above in connection with the description of prior figures.

Tick interrupt control circuit **880** generates a tick interrupt request on line **552** every 2.5 ms and monitors the acknowledgement of this interrupt request.

A positive edge detector generates a pulse every 2.5 ms. This sets a tick interrupt flip-flop, causing a tick interrupt request be asserted in the TINT REQ signal on line **552**. The pulse also causes a tick alarm flip-flop to sample the state of the tick interrupt flip-flop. If the tick interrupt flip-flop had not been cleared by the CLR TICK signal from the command decoder on line **882**, the tick alarm flip-flop is set to indicate that the previous tick interrupt request had not been processed in a timely manner. The tick alarm signal on line **872** disables all outputs of the PCC going to hardware devices, such as motors, print solenoids, etc. The TICK ALARM signal is also asserted by the RESET and CLK FALL signals, which indicate a failure of the basic clock signals of the PCC.

Clock divider **876** is a 7-bit synchronous binary counter used to generate the frequencies required by the BRM, baud rate generator, etc. A toggle accepts the 153.6 kHz frequency of the PH1 and PH2 clocks and divides it by a factor of 2 to produce a frequency of 76.8 kHz. This frequency is divided further by the up counters constituting the remaining six stages of the counter. The outputs F [0:6] provided on clock bus **874** are square wave signals that appear as the Q output of the stages of the divider. The output signal on line **884**, designated F0 PULSE, supplied to a divide-by-3 stage **890** is the carry output of the last stage of the clock divider and constitutes a pulse train with a pulse repetition frequency of 1200 Hz. The RESET signal on line **544** sets the divider to the 1111111 state.

Divide-by-3 stage **890** produces a 400 Hz signal on line **892** from the 1200 hertz signal of line **884**. The counting sequence is 00,01,10. The RESET signal sets the circuit to the 11 state, which is not reached in the counting operation.

High rate generator **900** and low rate generator **910** provide baud rate clocking signals at standardized baud rates by decoding the signals on the clock bus **874**. Baud output selector **920** provides as an output either the high baud rate from high rate generator **900** or a low baud rate from low rate generator **910**, the latter resulting from further division of the signals provided by the high rate generator. Baud command latch **930** receives baud rate selection commands over bus **64** and decodes them to control the selection of the baud rate clocks.

Bell controller **940** drives an audio alarm signal, such as a bell, to indicate various conditions which may be signalled to a user.

Line feed controller **950** generates the signal required by the line feed stepping motor. The desired state of the phase windings of that motor is stored under firmware control in internal latches. The desired state of the run transistor is similarly stored in a flip-flop. If the TICK alarm signal is asserted, the −R run output is disabled.

FIGS. 18A-18AR provides circuit diagrams of the apparatus blocks illustrated in FIGS. 16A-16D. Some of these circuit diagrams show circuit construction down to a gate level. In other cases, however, various functional circuit modules are shown. These modules are indicated by the designation LAxyz, where xyz indicates a three digit module type designation. Detailed circuit diagrams for each of these modules or cells are provided in FIGS. 19A(1)-19AB(2), together with truth tables where appropriate, to indicate the logic functions of the cells. (All FIGS. 19*(*) relate to the same module, where the * symbol indicates any letter or number.)

It is believed that the circuit diagrams for the apparatus blocks and the modular cells will both be readily understood without a detailed, gate-by-gate explanation, in view of the above detailed discussion of the operation of the PCC.

Nevertheless, to facilitate understanding of the circuit diagrams, the following brief explanations will be given for the circuit modules.

Circuit module LA400, FIGS. 19A(1)-19A(3), is a one-bit shift register for providing a delay function. Specifically, it is used where a 6.5 microsecond delay is needed, such as feedback shift register counting, edge detection, and other timing functions. The data input is simply propagated to the state output Q with a delay of one clock period. Optionally, the −Q terminal may also be used for the complementary state output. The PH1 and PH2 terminals receive those respective signals as clock inputs. At the PH2 time, the D input is sampled. At the PH1 time, the D input is propagated to the Q output and the inverse of the D sample is propagated to the −Q output.

The LA401 circuit module, shown in FIGS. 19B(1)-19B(3), implements a delay with a pre-set provision. The PH1 and PH 2 terminals receive those signals as clock inputs. The D terminal receives the data input, the Q terminal provides the state output and the S terminal is the set input. At the PH2 time, the D and S inputs are sampled. At the PH1 time, if the S sample is high, a high level is propagated to the Q output and a low level is propagated to the −Q output; otherwise, the D sample is propagated to the Q output and the inverse of the D sample is propagated to the −Q output.

Circuit module LA402, shown in FIGS. 19C(1)-19C(3), is a dynamic D-type flip-flop, also termed a sample-and-hold (S&H) circuit. The same terminology applies to its inputs, as above. In addition, there is an E terminal which receives an enable input. At time PH2, the D and E inputs, and the Q output are sampled. At the PH1 time, if the E sample is high, then the D sample is propagated to the Q output; otherwise, the Q output is merely refreshed.

Circuit module LA403, shown in FIGS. 19D(1)-19D(3), is a dynamic D-type flip-flop (i.e., S & H) with a set feature. Terminology is the same for its terminal connections. The R terminal, the use of which is optional, receives a reset input. At time PH2, the D, E, S, and R inputs are sampled, together with the Q output. At the PH1 time, if the S sample is high, the high level is propagated to the Q output; otherwise, if the R sample is high, then a low level is propagated to the Q output; otherwise, if the E sample is high, then the D sample is propagated to the Q output; otherwise, the Q output is merely refreshed.

Circuit module LA404, FIGS. 19E(1)-19E(2), is a data latchbit. Its L input terminal receives a load signal and its H terminal may be provided with a hold input. At time PH2, the D input and Q output are sampled. At time PH1, if the L input is high and the H input is low, then the D sample is propagated to the Q output. Otherwise, if the H input is high and the L input is low, then the Q output is refreshed. If both the H and L inputs are low, a low level is propagated to the Q output. Otherwise, with both H and L inputs high, if either the D sample or the Q sample is high, then the high level is propagated to the Q output.

Circuit cell LA 405, shown in FIGS. 19F(1)-19F(3), is a sampling filter. At time PH3, it samples the −D input and at the PH2 time, it samples both the −D input and −Q output. At time PH1, if any two of the PH3 and PH2 −D samples and −Q sample are low, then a low level is propagated to the −Q output; otherwise, a high level is propagated to the −Q output.

Circuit LA407, shown in FIGS. 19(G)1-19G(3), provides a toggle with set. At time PH2, the S input and the −Q output are sampled. At time PH1, if the S sample is high, then a high level is propagated to the Q output and a low level is propagated to the −Q output; otherwise, the −Q sample is propagated to the Q output and the inverse of the −Q sample is propagated to the −Q output.

Circuit module LA408, shown in FIGS. 19H(1)-19H(3), is a controlled toggle with provision for setting. The T input is the toggle input. At the PH2 time, the S input is sampled and if the T input is low, the Q output is sampled. Otherwise, the inverse of the Q output is sampled, thus forming a Q or −Q sample. At the PH1 time, if the S sample is high, then a high level is propagated to the Q output. Otherwise, the Q or −Q sample is propagated to the Q output. The result is that if the S input is low, then the Q output toggles at PH1 time only if the T input was high at PH2 time.

Circuit module LA409. FIGS. 19I(1)-19I(3), is an up counter with a set feature. The C terminal provides a carry output. This module is used as one stage of a binary up counter. If it is enabled, the output is toggled by the clock. A carry signal is presented to the next higher counter stage just before the output goes low. The carry signal is the enable signal of the next stage.

Asynchronously, if the T input is high, then the Q output is propagated to the C output. Otherwise, a low is propagated to the C output. At PH2 time, the S input is sampled and if the T input is low, the Q output is sampled. Otherwise, the inverse of the Q output is sampled, thus forming a Q or −Q sample. At PH1 time, if the S sampled is high, then a high level is propagated to the Q output. Otherwise, the Q or −Q sample is propagated to the Q output. The result is that if the S input is low, the Q output toggles at PH1 time only if the T input was high at PH2 time.

Circuit module LA410, shown in FIGS. 19J(1)-19J(2), is a data latch with an input clock interface. At the PH3 time the D input is sampled. At the PH2 time, the foregoing sample and the −Q output are sampled. At the PH1 time, if the L input is high and the H input is low, the inverse of the D sample is propagated to the −Q output. Otherwise, if the H input is high and the L input is low, then the −Q output is refreshed. If both the H and L inputs are low, a high level is propagated to the −Q output. Finally, if both H and L inputs are high, and either the D sample is high or the −Q sample is low, then a low level is propagated to the −Q output.

Circuit module LA411, shown in FIGS. 19K(1)–19K(3), is an up/down counter with a set feature. The U input provides up/down control. Asynchronously, if both the T and U inputs are high, then the Q output is propagated to the C output. However, if the T input is high and the U input is low, then the inverse of the Q output is propagated to the C output. If the T input is low, a low level is propagated to the C output. At PH2 time, the S input is sampled and, if the T input is low, the Q output is sampled; otherwise, the inverse of the Q output is sampled, thus forming a Q or −Q sample. At PH1 time, if the S sample is high, then a high level is propagated to the Q output; otherwise, the Q or −Q sample is propagated to the Q output. The result is that if the S input is low, the Q output toggles at PH1 time, but only if the T input was high at PH2 time.

Circuit module LA412, shown in FIGS. 19L(1)–19L(2), is a write pulse generator. At PHX time, if the S input is low, then internal node A is sampled, thus forming an A sample; unless the −S input is low, in which case the A sample is forced high; unless the R input is high, in which case the A sample is forced low. At PHY time, the A sample is propagated to internal node A. At the coincidence of PHX time and PH1 time, the S input and internal node A are sampled, and if either of these samples is high, a low level is propagated to the −Q output; otherwise, a high level is propagated to the −Q output.

Circuit module LA413, shown in FIGS. 19M(1)–19M(3), is a dynamic R/−S type flipflop. At PH2 time, the R input, the −S input and −Q output are sampled. At PH1 time, if the −S sample is low, a low level is propagated to the −Q output. Otherwise, if either the R sample or the −Q sample is high, a low level is propagated to the −Q output. If both the −Q sample and R sample are low, a low level is propagated to the −Q output. The inverse of the −Q output is asynchronously propagated to the Q output, if present.

Circuit module LA414, shown in FIGS. 19N(1)–19N(3), is a fifo controller used in the dot buffers. Each such circuit controls one column in the dot fifo. It generates the necessary signals so that the fifo column can accept dot data from the preceding column and shift it to the next column as required.

The Ri terminal is for a ripple input and, likewise, the Ro terminal, which is optional, is for a ripple output. The I terminal is for an initialization input. The −M terminal provides a match (i.e., compare) output.

Asynchronously, the Ro output, if present, is low, unless the Set input is high and either the input Ri or the −Q signal or both Ri and −Q are high. At PH2 time, the inverse of the Q output is sampled, thus forming a −Q sample, unless either or both the Ri and S inputs are high. If S is high, the −Q sample is forced low regardless of the state of Ri. If Ri alone is high, the −Q sample is forced high. Also at PH2 time, if −Q, Ri, and the −Q sample are all low, a high level is propagated to the L output; otherwise, a low level is propagated. The inverse of the L output is propagated to the H output. At PH1 time, the inverse of the −Q sample is propagated to the Q output, provided that the I input is low. If the I input is high, Q is forced low.

The LA415 circuit module, FIGS. 19O(1)–19O(2), is an up/down counter with load, reset and compare capability. The C terminal provides an optional carry/borrow output and the U terminal provides an optional up/-down control input.

Asynchronously, if the U input and Q output are both high, or the U input and Q output are both low, then the T input is propagated to the C output, if present. Also asynchronously, if the D input and the Q output are both high, or the D input and Q output are both low, then a low level is propagated to the −M output.

At PH2 time, the inverse of the Q output is sampled, unless the T and L inputs are both low, in which case the Q output is sampled, thus forming a Q or −Q sample. However, if the −L input is high, the Q or −Q sample is forced to the level of the D input. But if the R input is high, the Q or −Q sample is forced low. At PH1 time, the Q or −Q sample is propagated to the Q output.

The LA418 circuit module, shown in FIGS. 19Q(1)–19Q(3), is an up counter with load and reset features. Asynchronously, if the Q output is high, then the T input is propagated to the C output. At PH2 time, if the T input is high, then the Q output is sampled; or if the T and L inputs are both low, then the inverse of the Q output is sampled, thus forming a Q or −Q sample. But if the −L input is low, the Q or −Q sample is forced to the level of the D input; and if the R input is high, the Q or −Q sample is forced low. At PH1 time, the Q or −Q sample is propagated to the Q output.

The LA419 cell, shown in FIGS. 19Q(1)–19Q(3), is a timer controller. The S terminal provides a start input and the R terminal provides a run input.

$$L = E \text{ and } (S \text{ and } Q \text{ or } -R \text{ and } (S \text{ or } Q)).$$

$$Q[n+1] = E[n] \text{ and } (S[n] \text{ and } Q[n] \text{ or } R[n] \text{ and } (S[n] \text{ or } Q[n]))$$

Asynchronously, if the −R input is high and either the Q output or the S input is high, or the Q output and S input are both high, regardless of the −R input state, then the inverse of the −E (enable) input is propagated to the L output. Otherwise, a low level is propagated to the L output. At PH2 time, if the R input is high and either the Q output or the S input is high, or the Q output and S input are both high (regardless of R input state), then the inverse of the −E input is sampled, thus forming an E sample, otherwise the E sample is forced low. At PH1 time, the E sample is propagated to the Q output.

The LA421 circuit module, depicted in FIGS. 19R(1)–19R(2), is a positive edge detector. The X terminal receives data input and the −Y terminal provides a −edge output. The complimentary, Y, output is optional.

At PH2 time, the X input is sampled. At PH1 time, if the X sample is high, then the inverse of the X input is propagated to the −Y output. The inverse of the −Y output is propagated to the Y output, if present.

Circuit module LA423, shown in FIGS. 19S(1)–19S(3), is a dynamic R/S flip-flop. S1 and S2 are separate set inputs. At PH2 time, the Q output is sampled, unless the R input is high, in which case the Q output is forced low. But if the S1 or S2 input is high, the Q sample is forced high. At PH1 time, the Q sample is propagated to the Q output and the inverse of the Q sample is propagated to the −Q output.

Circuit cell LA424, FIGS. 19T(1)–19T(2), is a data latch controller. At PH2 time, the E input is propagated to the L output and the inverse of the E input is propagated to the H output.

Circuit module LA425, FIGS. 19U(1)–19U(2), is a dual input data latch bit. At PH2 time, the D1, D2, and S inputs (if present) and Q output are sampled. At PH1 time, if the L1 input is high, then the D1 sample is propagated to the Q output. If the L2 input is high, the D2 sample is propagated to the Q output. If the H input is high, then the Q output is refreshed. Otherwise, a low level is propagated to the Q output. If more than one of the three inputs L1, L2 and H are high, then any attempt to propagate a high to the Q output will override any attempt to propagate a low.

Circuit module LA428, shown in FIGS. 19V(1)–19V(2), is a controlled toggle with enable and reset. At PH2 time, the R1 and R2 inputs are sampled. If the E or T input is low, then the $-Q$ output is sampled; otherwise, the inverse of the $-Q$ output is sampled, forming a $-Q$ or Q sample. At PH1 time, if the R1 sample or the R2 sample is high, then a high level is propagated to the $-Q$ output; otherwise, the $-Q$ or Q sample is propagated to the $-Q$ output. The result is that if the R1 input and R2 input are both low, then the Q output toggles at PH1 time only if the T input and E input were both high at PH2 time.

Module LA434, as shown in FIGS. 19W(1)–19W(3), is a data latch controller with reset capability. At PH2 time, the E input is propagated to the L output and the inverse of the E input is propagated to the H output, unless the R input is high, in which case the H output is forced low.

Circuit cell LA450, shown in FIGS. 19X(1)–19X(2), is a four bit shifter. DA is a data input terminal, QA–QD are state outputs, SB and SC are set inputs, M is a maximum output, Z is a zero output and R is a reset input.

Circuit module LA451, FIGS. 19Y(1)–19Y(2), is a clock generator and divide-by-13 circuit. It generates the PH1-PH4 clock outputs from the PHX and PHY clock inputs. Zero and reset inputs are also provided.

Circuit module LA500, shown in FIGS. 19Z(1)–19Z(2), is a clock driver and clock failure detector. It receives a clock input C, provides the PHX and PHY clock outputs and a clock fail output, F.

Circuit module LA410, shown in FIGS. 19AA(1)–19AA(2) is a 1-of-4 selector. It has two address inputs, A0 and A1, and four selection outputs, S0–S3.

Circuit module LA600, shown in FIGS. 19AB(1)–19AB(2), is a static D-type flip-flop.

What is claimed as new and desired to be secured by Letters Patent of the United States is:

1. A control system for a dot matrix character printer for printing characters of selectable pitch, which includes a print head containing at least one column of dot printing means, means for moving the print head relative to a medium on which characters are to be printed, and transition generator means for providing print head motion signals responsive to movement of the print head, each basic unit of print head displacement defining a transition, the control system comprising:

sequential controller means for providing signals to effectuate sequential operation of portions of the control system;

the sequential controller means providing a request-to-print signal indicating the controller's readiness to begin supplying sequential signals for printing a string of columns;

print head actuation control means for providing print head actuation signals for actuating the dot printing means;

the print head actuation control means being adapted to enable printing responsive to the concurrence of the request-to-print signal and a print head motion signal indicating arrival of the print head at the location where the column of dot printing means is to be actuated for printing the first column of a string of columns, such location being one of a multiplicity of predetermined locations; and

the sequential controller means being adapted to provide, and the print head actuation control means being further adapted to accept, a request-to-print signal independent of the position of the print head, whereby a request-to-print signal can be provided at any point to initiate printing at the next one of said predetermined locations reached by the moving print head.

2. The dot matrix character printer control system of claim 1 wherein said multiplicty of predetermined locations are determined by the most recent selection of character pitch.

3. The dot matrix character print control system of claim 1 wherein the location of said concurrence is a function of print head velocity, whereby compensation may be provided for movement of the print head during the time interval between actuation of the dot printing means and the printing of dots, permitting character printing at variable speeds of print head motion.

4. The dot matrix character printer control system of claim 3 wherein the print head actuation signals for the printing of a character are provided only during the interval in which the print head actuation control means enables printing.

5. The dot matrix character printer control system of any of claims 1, 3 or 4 wherein the print head may move in two directions, such as left-to-right and right-to-left, and further including:

means for signalling in which of said two directions the print head is moving; and

responsive to the means for signalling, the first column of a character being set to be the left-most column of the character for printing from left-to-right and being set to be the right-most column of the character for printing from left-to-right.

6. A dot matrix character printer for printing characters of a pitch selectable from among a plurality of available pitches, comprising:

a print head;

means for moving the print head;

transition generator means for providing print head motion signals responsive to movement of the print head, each basic unit of print head displacement defining a transition;

transition counter means for providing print head actuation signals responsive to the print head motion signals, the distance travelled by the print head between successive print head actuation signals defining an increment;

increment counter means for providing intercharacter space signals responsive to print head actuation signals;

said transition counter means and said increment counter means being variable radix counters;

means for accepting signals representing the number of increments per character, the number of transitions per increment and the number of transitions per intercharacter space;

means for setting the radix of the increment counter means to the number of increments per character represented by said signals;

means for setting the radix of the transition counter means, such radix setting means being adapted to set the radix of the transition counter means to the number of transitions per intercharacter space during the intercharacter space and to the number of transitions per increment at all other times; and

the increment counter means and the transition counter means being adapted to accept as their radices any values within their respective counting ranges.

7. The dot matrix character printer of claim 6 further including means for setting the radix of the increment counter means to the number of increments per character.

8. The dot matrix character printer of claim 6 further including means for storing a representation of the number of increments per character, the number of transitions per increments and the number of transitions per intercharacter space, responsive to the means for accepting, and the radix setting means further being adapted to selectively provide such stored representations of the number of transitions per increment and transitions per intercharacter space to the transition counter for setting its radix.

9. The dot matrix character printer of either claim 7 or claim 8 wherein the means for setting the radix of the transition counter and the means for setting the radix of the increment counter are adapted to permit the number of increments per character, the number of transitions per increment and the number of transitions per intercharacter space to vary from character to character, thereby permitting character pitch to be altered on a character by character basis.

10. The dot matrix character printer of claim 9 wherein the print head may move to two directions, such as left-to-right and right-to-left, and further including:

means for signalling in which of said two directions the print head is moving; and

responsive to the means for signalling, the first column of a character being set to be the left-most column of the character for printing from left-to-right and being set to be the right-most column of the character for printing from left-to-left.

11. A dot matrix character printer for printing characters of a pitch selectable from among a plurality of available pitches, comprising:

a print head including dot printing means for printing dots to form characters;

means for moving the print head;

transition generator means for providing print head motion signals in response to movement of the print head, the basic unit of print head displacement resolved by the print head motion signals defining a transition;

transition counter means for counting transitions to provide print head actuation signals in response to the print head motion signals, the distance travelled by the print head between successive print head actuation signals defining an increment;

increment counter means for counting increments to provide intercharacter space signals;

said transition counter means and said increment counter means comprising variable radix counters;

means for accepting signals representing the number of transitions per increment and the number of transitions per intercharacter space;

means responsive to said intercharacter space signals for setting the radix of the transition counter means to the number of transition per intercharacter space during the intercharacter space and to the number of transitions per increment at all other times, to generate an intercharacter space having at least one transition independent of the number of transitions per increment;

means for setting the radix of the increment counter means to the number of increments per character;

the increment counter means and the transition counter means being adapted to accept at their radices any values within their respective counting ranges; and

the provision of print head actuation signals being inhibited at selected times, so that characters are printed on an increment by increment basis, with selected pitch.

12. The dot matrix character printer of claim 11 further including:

transition counter control means for causing the transition count in the transition counter means to be incremented or decremented responsive to a transition add/subtract signal, whereby the print head actuation position may be controlled by the transition add/subtract signal to compensate for movement of the print head during the response time of the dot printing means in the print head and permitting accurate registration of printed characters while printing at variable rates.

13. The dot matrix character printer of either claim 11 or claim 12 further including:

net transition generator means associated with the transition counter for preventing the transition counter from responding to pairs of consecutive single transitions of opposite directions, such net transition generator means including:

(a) means for comparing a subject transition with the next preceding transition;

(b) inhibiting means responsive to the means for comparing, for preventing the transition counter from responding to the subject transition if it represents a reversal of direction from the next preceding transition;

(c) means responsive to the inhibiting means, for storing the direction of the subject transition and the fact that the transition counter was prevented from responding to it;

(d) second comparison means for comparing the next succeeding transition with the subject transition; and

(e) means responsive to the second comparison means, for preventing the transition counter from responding to such next succeeding transition if the directions of the subject transition and the next succeeding transition are different and for providing both transitions to the transition counter if such directions are the same.

14. The dot matrix character printer of claim 11 wherein the print head includes two sets of dot printing means, the dot printing means of each set being actuable in response to print head actuation signals supplied to the dot printing means, the print head actuation signals provided by the transition counter means defining primary increment signals, and further including:

primary increment signal storage means responsive to the primary increment signals for producing secondary increment signals, each of said secondary increment signals being displaced with respect to a corresponding primary increment signal by an amount corresponding to the displacement between sets of dot printing means; and

a first one of said sets of dots printing means being actuated by the primary increment signals and the other, second set of dot printing means being actuated by the secondary increment signals.

15. The dot matrix character printer of claim **14** wherein the print head may travel in two directions; and according to the direction of travel of the print head, one set of dot printing means is designated the leading set and the other set of dot printing means is designated the trailing set, the leading set being that set of dot printing means which leads in the direction of travel of the print head; and further including:

means for providing a direction signal representative of the direction of print head travel; and

means responsive to the direction signal for providing the primary increment signal to the leading set of dot printing means and the secondary increment signal to the trailing set of dot printing means;

whereby characters may be printed with selectable and varying pitches and at variable print head velocities with a print head having two sets of dot printing means while the print head moves to each of said two directions.

16. The dot matrix character printer of claim **14** or claim **15** wherein the primary increment signal storage means for producing secondary increment signals comprises a shift register clocked by print head motion signals and receiving at its signal input the primary increment signals, such shift register producing as its output signal the secondary increment signals, the shift register length in bits being equal to the displacement in transitions between the sets of dot printing means, whereby each secondary increment signal is a displaced counterpart of a primary increment signal, the displacement between them corresponding to the displacement between sets of dot printing means.

17. The dot matrix character printer of any of claims **11**, **14** or **15** wherein the dot printing means comprise solenoid driven print wires for impact printing, and the response time of the dot printing means includes the flight time of such print wires.

18. In a dot matrix character printer of the type having a print head positioned by a motor and encoder means for producing an encoder signal responsive to the incremental changes in print head position, said encoder signal alternating between two states and changing state when the print head moves a predetermined distance, the improvement comprising:

encoder signal filter means operable to remove relatively short duration noise signals from the encoder signal and to thereby provide a filtered encoder signal,

the encoder filter means comprising means for sampling the encoder signal a plurality of times during the smallest increment of print head movement of interest and for changing the state of the filtered encoder signal only when the encoder signal is in the same state for a majority of such sample times.

19. A printer control circuit for use in a dot matrix character printer for printing characters of a pitch selectable from among a plurality of available pitches, said printer being of the type having a print head which includes dot printing means for generating dots, means for moving the print head, and encoder means for generating signals indicative of relative print head position, the signals generated by the encoded being referred to as transition signals and the smallest unit of print head displacement resolved by the transition signals being referred to as a transition, the control circuit comprising:

transition counter means for counting transitions to provide print head actuation signals in response to the transition signals, the distance travelled by th print head between successive print head actuation signals defining an increment;

increment counter means for counting increments to provide intercharacter space signals during the intercharacter space, which is defined as a buffer zone following a character and during which printing of another character is prohibited;

said transition counter means and said increment counter means comprising variable radix counters;

means for accepting a signal representing the number of increments per character, the number of transitions per increment and the number of transitions per intercharacter space;

means for initializing the transition counter means and the increment counter means;

means responsive to said intercharacter space signals for setting the radix of the transition counter means to the number of transitions per intercharacter space in the intercharacter space and to the number of transitions per increment at all other times, to generate an intercharacter space having at least one transition independent of the number of transitions per increment;

means for setting the radix of the increment counter means to the number of transitions per character;

the increment counter means and the transition counter means being adapted to accept as their radices any values within their respective counting ranges; and

means for controlling the position where the printing of a character is to start, including means responsive to the state of a two-state character print signal for inhibiting the provision of print head actuation signals to the print head until such character print signal occurs in a first state, establishing a window during which printing may start, and simultaneously the generation of an intercharacter space is completed, as signified by the intercharacter space signal.

20. The dot matrix character printer control circuit of claim **19** further including:

transition counter control means for causing the transition count in the transition counter means to be incremented or decremented responsive to a transition add/subtract signal, whereby the print head actuation position may be controlled by the transition add/subtract signal to compensate for movement of the print head during the response time of the dot printing means in the print head and permitting accurate registration of printed characters while printing at varible rates.

21. The dot matrix character of either claim **19** or **20** further including:

net transition generator means associated with the transition counter for preventing the transition counter from responding to pairs of consecutive

single transitions of opposite directions, such net transition generator means including:

(a) means for comparing a subject transition with the next preceding transition;

(b) inhibiting means responsive to the means for comparing, for preventing the transition counter from responding to the subject transition if it represents a reversal of direction from the next preceding transition;

(c) means responsive to the inhibiting means, for storing the direction of the subject transition and the fact that the transition counter was prevented from responding to it;

(d) second comparison means for comparing the next succeeding transition with the subject transition; and

(e) means responsive to the second comparison means, for preventing the transition counter from responding to such next succeeding transition if the directions of the subject transition and the next succeeding transition are different and for providing both transitions to the transition counter if such directions are the same.

22. The dot matrix character printer of claim 21 wherein the dot printing means comprise solenoid driven print wires and the response time of the dot printing means includes the flight time of such print wires.

* * * * *

United States Patent [19]

Hiroichi et al.

[11] **Patent Number: 4,459,431**

[45] **Date of Patent: Jul. 10, 1984**

[54] **OUTPUT CONTROL SYSTEM FOR DOT-TYPE PRINTER**

[75] Inventors: **Teramura Hiroichi, Tokyo; Yasuhiro Yamazaki, Hiratsuka; Kazuo Hasuike, Tokyo, all of Japan**

[73] Assignee: **Kokusai Denshindenwa Co., Ltd., Tokyo, Japan**

[21] Appl. No.: **402,571**

[22] Filed: **Jul. 28, 1982**

[30] **Foreign Application Priority Data**

Jul. 31, 1981 [JP] Japan 56-119272

[51] Int. Cl.³ H04L 15/24

[52] U.S. Cl. 178/30; 400/121; 400/124

[58] Field of Search ..... 178/30, 26 R; 340/735; 400/124, 121

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,999,168 12/1976 Findley et al. 178/30
4,298,290 11/1981 Barnes et al. 400/3

OTHER PUBLICATIONS

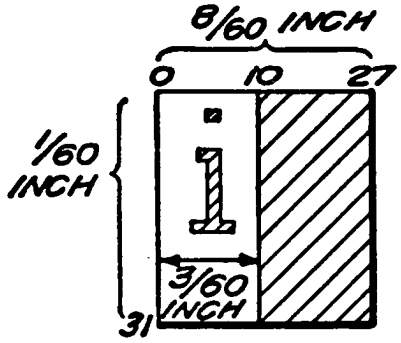
A. J. Daflib & R. R. Schomburg, Proportional Spacing Roster Printer Character Generator, IBM Technical Disclosure Bulletin, vol. 15, No. 8, (Jan. 1973), pp. 2215–2216.

Primary Examiner—Stafford D. Schreyer

[57] **ABSTRACT**

A dot-type printer has heretofore been unable to output characters of the pitch which does not coincide with multiples of the dot-width by integers when such dot-width is either electrically or mechanically fixed, causing considerable errors and not being in compliance with the recommendation of CCITT. The system according to the present invention provides a means to obtain the distance between a character and a printing reference point and to output the character at a position which is separated from the reference point by the distance in a multiple of the dot-width by an integer and which is either immediately before or after the point of the distance thus obtained. By such output control, the character outputted can remain within the range recommended by CCITT with only negligible errors.

3 Claims, 9 Drawing Figures



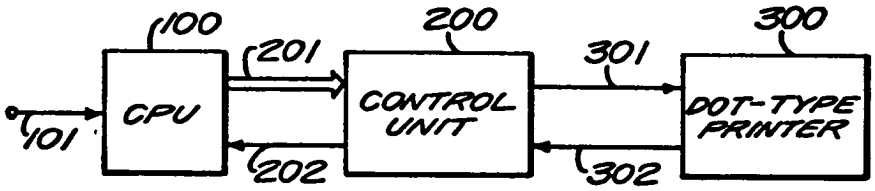

100
200
300
201
301
CPU
CONTROL UNIT
DOT-TYPE PRINTER
101
202
302

FIG. 1

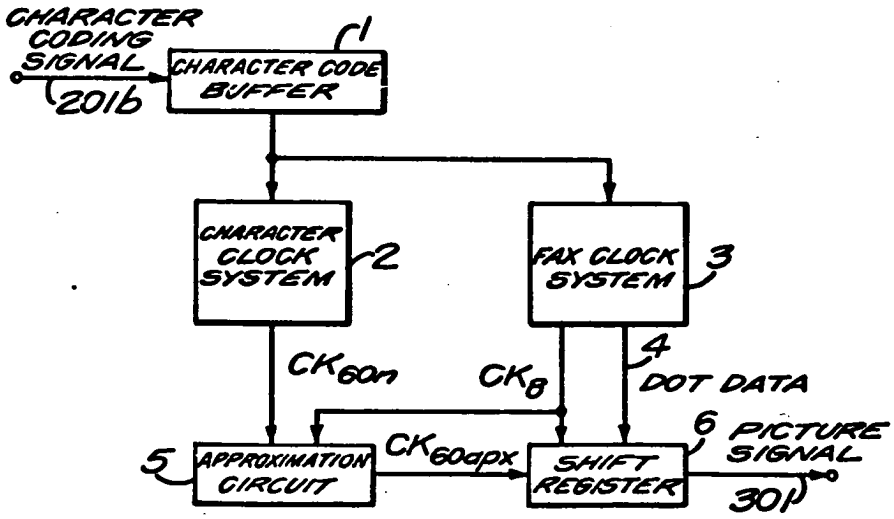

CHARACTER CODING SIGNAL
2016
1
CHARACTER CODE BUFFER
CHARACTER CLOCK SYSTEM
2
FAX CLOCK SYSTEM
3
CK60n
CK8
4
DOT DATA
6
PICTURE SIGNAL
5
APPROXIMATION CIRCUIT
CK60apx
SHIFT REGISTER
301

FIG. 2

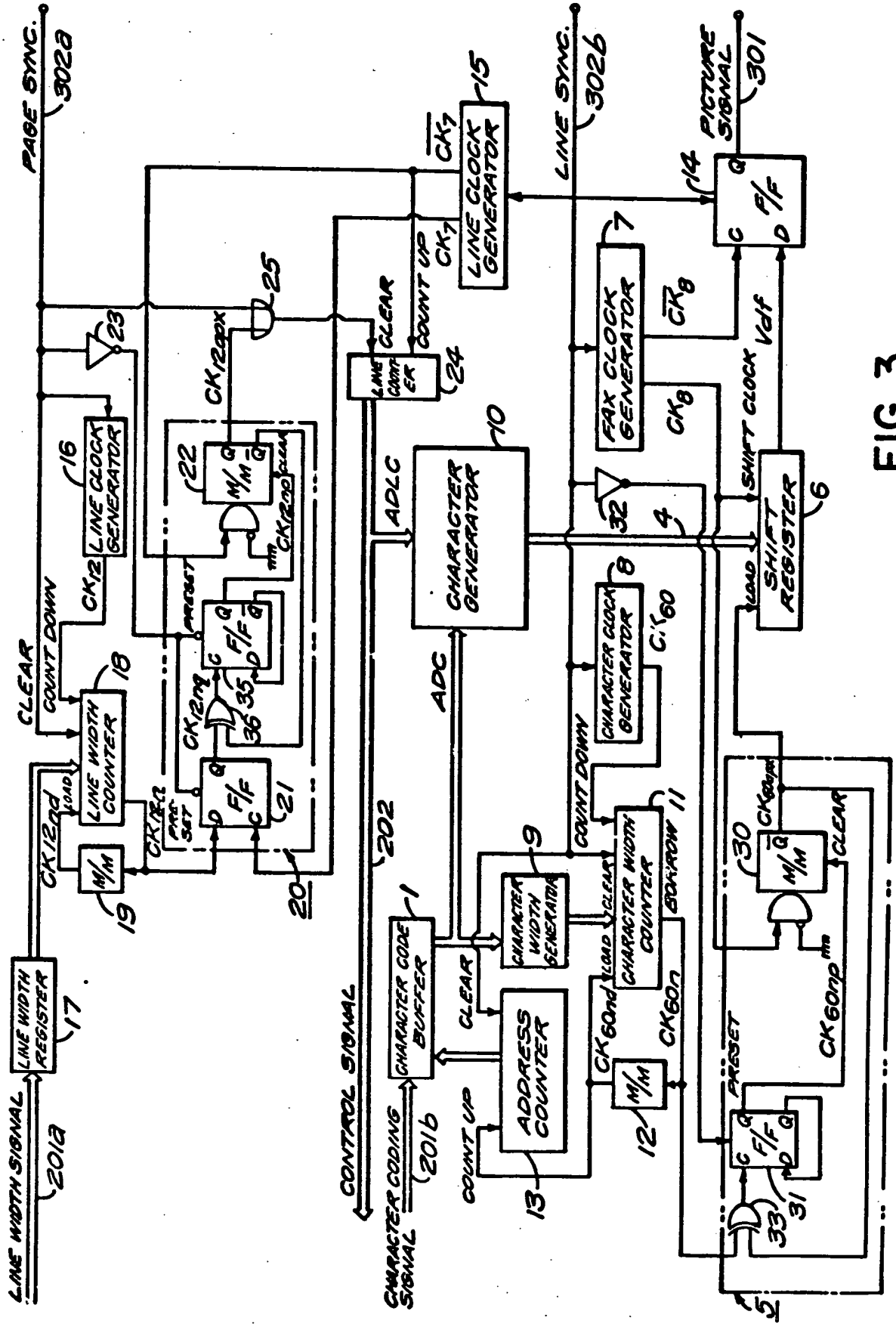

PAGE SYNC.
LINE SYNC.
PICTURE SIGNAL
LINE WIDTH SIGNAL
CHARACTER CODING SIGNAL
CONTROL SIGNAL
LINE WIDTH REGISTER
LINE CLOCK GENERATOR
LINE WIDTH COUNTER
CHARACTER GENERATOR
CHARACTER CODE BUFFER
CHARACTER WIDTH GENERATOR
CHARACTER WIDTH COUNTER
CHARACTER CLOCK GENERATOR
ADDRESS COUNTER
FAX CLOCK GENERATOR
SHIFT REGISTER
COUNT DOWN
COUNT UP
CLEAR
LOAD
PRESET
ADC
ADLC
SHIFT CLOCK
F/F
M/M

FIG. 3

FIG. 4(a) PAGE SYNC. 302a

FIG. 4(b) LINE SYNC. 302b

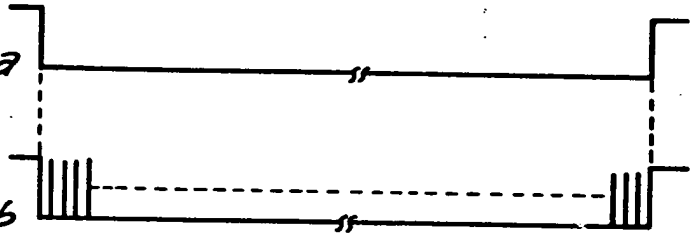


FIG. 5(a) PAGE SYNC. 302a

FIG. 5(b) LINE SYNC. 302b

FIG. 5(c) FAX CLOCK CK_8

FIG. 5(d) CHARACTER CLOCK CK_{60}

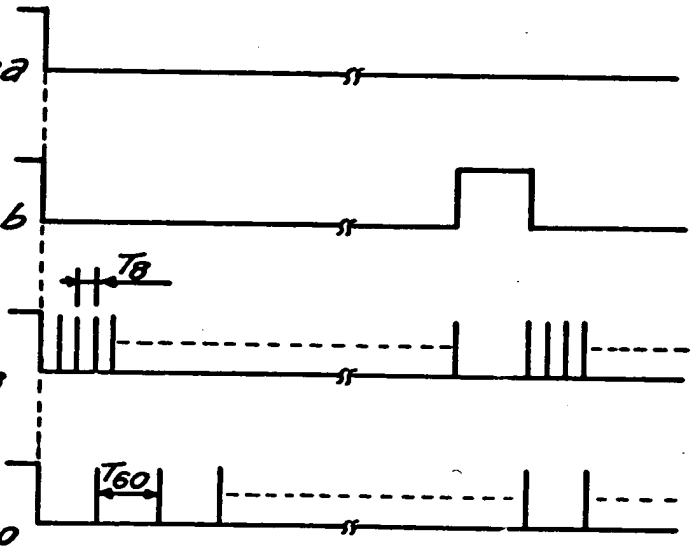


FIG. 6(a)

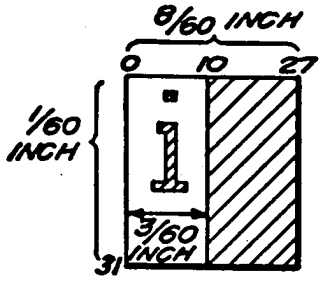


FIG. 6(b)

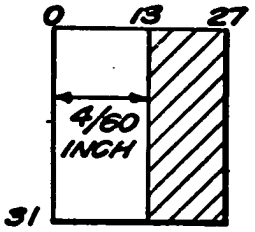


FIG. 6(c)

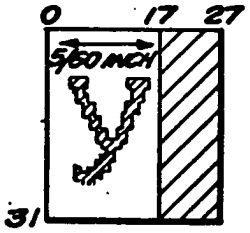


FIG. 6(d)

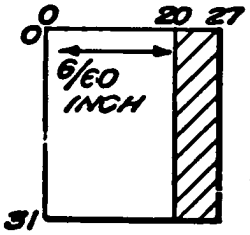


FIG. 6(e)

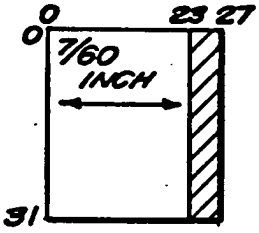
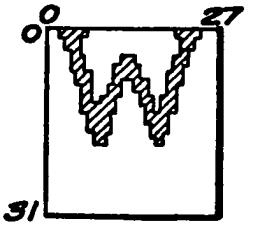


FIG. 6(f)



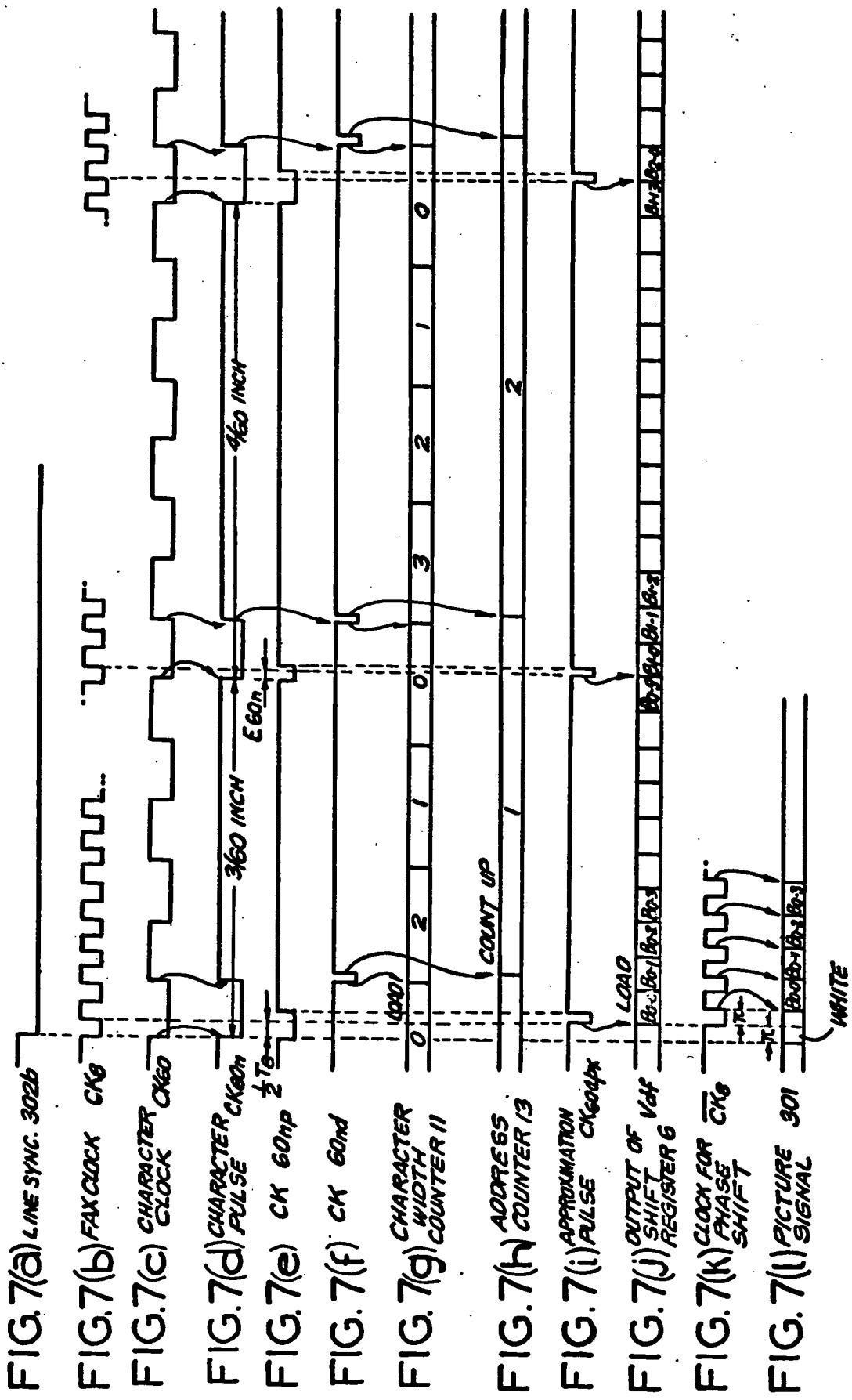


FIG. 7(a) LINE SYNC. 302a
FIG. 7(b) FAX CLOCK CKo
FIG. 7(c) CHARACTER CLOCK CKeo
FIG. 7(d) CHARACTER PULSE CKeon
FIG. 7(e) CK 60nd
FIG. 7(f) CK 60nd
FIG. 7(g) CHARACTER WIDTH COUNTER 11
FIG. 7(h) ADDRESS COUNTER 13
FIG. 7(i) APPROXIMATION PULSE CK604fx
FIG. 7(j) OUTPUT OF SHIFT REGISTER 6 Vdf
FIG. 7(k) CLOCK FOR PHASE SHIFT CKo
FIG. 7(l) PICTURE SIGNAL 301
360 INCH
460 INCH
COUNT UP
LOAD
WHITE

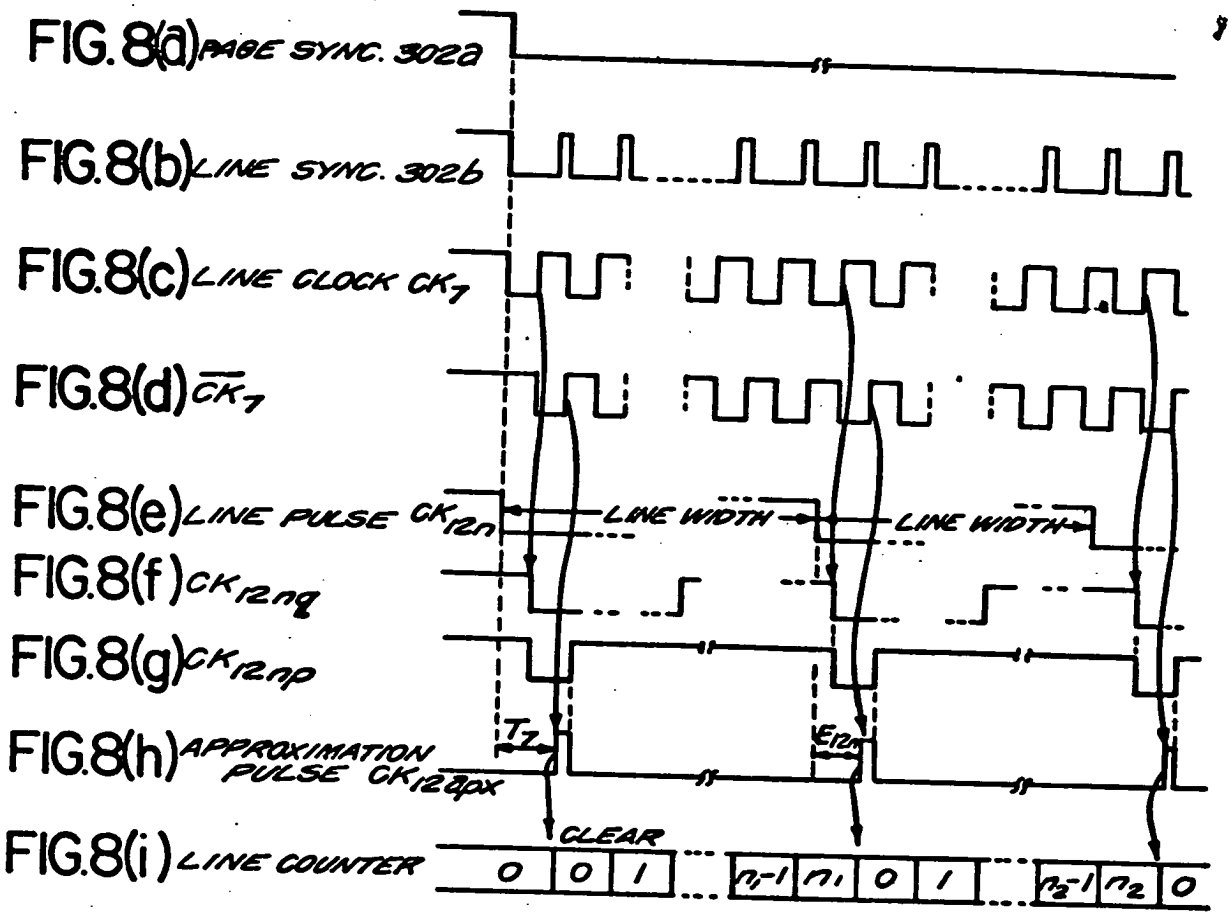


FIG. 8(a) PAGE SYNC. 302a
FIG. 8(b) LINE SYNC. 302b
FIG. 8(c) LINE CLOCK CK_7
FIG. 8(d) $\overline{CK_7}$
FIG. 8(e) LINE PULSE CK_{12n}
LINE WIDTH
LINE WIDTH
FIG. 8(f) CK_{12n_2}
FIG. 8(g) CK_{12n_1}
FIG. 8(h) APPROXIMATION PULSE CK_{12qox}
T_Z
ε_{12n}
CLEAR
FIG. 8(i) LINE COUNTER
0 0 1
n_1-1 n_1 0 1
n_2-1 n_2 0

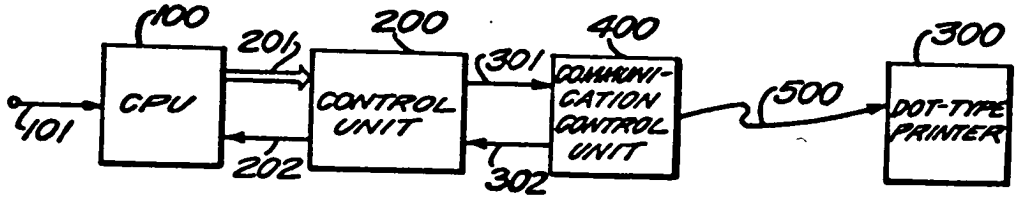

100
CPU
101
201
202
200
CONTROL UNIT
301
302
400
COMMUNI-CATION CONTROL UNIT
500
300
DOT-TYPE PRINTER

FIG. 9

OUTPUT CONTROL SYSTEM FOR DOT-TYPE PRINTER

BACKGROUND OF THE INVENTION

The present invention concerns an output control system which is capable of having a dot-type printer effectively output objects such as characters with a pitch which does not coincide with a multiple of the dot-width by an integer which said device either electrically or mechanically is inherently provided with.

A dot-type printer comprises a system to scan an object such as a printing paper in the lateral direction (which is generally referred to as a main scanning system) and a system to scan it in the vertical direction (which is generally referred to as a sub-scanning system) and outputs dots at an interval either mechanically or electrically predetermined so as to express a character or a diagram with assembled dots. The printer of above mentioned type has been put into practice as an output system for various data processing and data transmission systems such as the facsimile communication or the character communication.

As widely known, a facsimile communication is a system wherein the data to be transmitted such as manuscripts or original drawings are divided into predetermined picture elements, the system being particularly appropriate for transmitting drawings. A character communication is the system wherein a text consisting of characters alone is converted into codes by each character for transmission which are subsequently decoded back to characters on the receiving side.

There is no problem in the case such a dot-type printer is used as an output device separately for various individual communication systems such as the facsimile system, the character communication system, etc.

The intercommunication system between different communication terminals has become possible in recent years as data communication technology progressed; for example, characters can be outputted from a computer by a character communication to a facsimile receiver. Since the coding efficiency is predominantly higher in the character coding than the facsimile coding, there has been contemplated a composite communication system wherein the character data contained in a manuscript to be transmitted is coded by the character coding while the drawing data therein is coded by the facsimile coding for transmission.

In the cases of intercommunication or composite communication mentioned above however, there is involved a basic problem in that a single dot-type printer has to cope with outputs of both character communication and facsimile communication. More specifically the problems can be summarized as below:

(1) The facsimile resolution is determined by CCITT (Internation Telegraph and Telephone Consultive Committee) at 8.04 pel. (picture element)/mm in the direction of main scanning while the linear density (number of lines) is determined at 7.7 lines/mm or 3.85 lines/mm in the direction of sub-scanning.

(2) As to characters, according to CCITT a pitch (a space allocated for a character) is set at either one of the multiples of 2.54 mm/character (1/10 inch/character) for the lateral direction and of 4.233 mm/line (1/6 inch/line) for the vertical direction by one of the integers of 0.5, 1, 1.5 or 2. For other characters including proportional spacing characters traditionally used, the intervals are determined at a multiple of 1/60 inch by an integer (by 2 to 8) in the lateral direction and at a multiple of 1/12 inch by an integer (by 1 to 4) in the vertical direction.

(3) As shown in Tables 1 and 2, the character pitches above mentioned do not coincide with the multiple of the dot-width by any one of the integers both in lateral and vertical directions of facsimile, deviating from the range.

(4) In prior art, pitches of characters were fixedly approximated to a multiple of the facsimile dot-width by an integer. That, however, causes errors; for instance, in the case where the character width of 1/10 inch (2.54 mm) is approximated to 20 pel against 20.41 pel of facsimile dot-width, the error of about 2% occurs in accumulation, shortening the print of 200 mm by 4 mm or 2 characters. The pitches are heretofore not in compliance with the recommendation of CCITT.

TABLE 1

Character Width (inch)	Number of Pels in Facsimile
2/60	6.80
3/60	10.21
4/60	13.61
5/60 (1/12)	17.01
6/60 (1/10)	20.41
7/60	23.82
8/60	27.22

TABLE 2

Line Width (inch)	Number of Scanning Lines in Facsimile (In the Case of 7.7 lines/mm)
1/12	16.30
1/6	32.60
½	48.89
⅔	65.19

SUMMARY OF THE INVENTION

The present invention aims to solve aforementioned problems encountered in prior art and to provide an output control system which is capable of outputting characters with pitches which does not coincide with the multiples of dot-width by an integer which the dot-type printer either electrically or mechanically is provided with, only within minor errors, but in compliance with the CCITT recommendation for example.

In order to achieve such a purpose, the present invention is characterized in that when an output object having a pitch different from a multiple by an integer of a dot-width is outputted to a dot-type printer having a constant resolution either in main scanning or in sub-scanning directions, the distance from a reference point is obtained for respective output objects, the positions of respective objects are approximated to a position either immediately before or after the said obtained distance among the positions separated from the reference point by the distances which are the multiples of dot-width by integers.

The present invention is based upon a technical concept that at the time of printing, the present position from a reference point (such as the left side of the character frame of the first character or the home position) is always made to be recognized, and if there is an error, the position is approximated to a dot position which is the closest position to the correct one when viewed from the reference point. The present invention is applicable to both character pitches in the main scanning direction (character width) and in the sub-scanning

direction (line width) and further applicable not only to the case wherein character coding takes place at the same place where the dot-type printer exists but also to the case where two systems are located at separate and remote places and character codes are converted in facsimile data on the side of transmission to be transmitted via a communication circuit to a dot-type printer.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram to show an embodiment according to the present invention.

FIG. 2 is a schematic view to show a control system which embodies the present invention.

FIG. 3 is a circuit diagram to show an embodiment of the control system.

FIGS. 4, 5, 7 and 8 are timing charts to explain the performances of FIG. 3.

FIG. 6 is an explanatory view of a character generator.

FIG. 9 is a block diagram of another embodiment of the present invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The present invention will now be explained in detail referring to embodiments shown in attached drawings. Since dot-type printer devices can be represented by a facsimile receiver, a facsimile receiver is used hereinafter for the sake of explanation.

FIG. 1 shows an embodiment of the present invention wherein the reference numeral 100 denotes a central data processor unit (which is referred to as CPU hereinafter), 200 a control unit which embodies the present invention, and 300 a dot-type printer. FIG. 2 shows schematically an embodiment of the control unit 200.

In FIG. 1 CPU 100 stores one page of the data 101 which is inputted by an input means such as a communication circuit, etc. and they generate a character line for a line by inserting an appropriate number of blank signals between the start point and the home position of the dot-type printer 300 or adding such in the end of a line if the line is too short to composite a full line. CPU 100 receives a control signal 202 from the control unit 200 and transmits a print signal 201 including character coding signals to express the character line for said lines and signals to express the interval for lines to the control unit 200.

The resolution or the dot-width of the dot-type printer 300 is fixed either mechanically or electrically both in main scanning and the sub-scanning directions. In this case the resolution is 8.04 pel/mm in the main scanning direction and 7.7 lines/mm in the sub-scanning direction.

The control unit 200 which embodies the present invention controls the character code lines from CPU 100 by using a synchronizing signal 302 from the printer 300 and transmits a picture signal 301 to the dot-type printer 300. The picture signal is the one outputted in the form of a bit parallel row for scanning line.

The control unit 200 is provided with a function which can obtain the position at least for one scanning direction or the distance from a reference point (e.g. the home position or the top of a page) for respective objects (such as characters or signals) of which line width and line interval are not coincided with multiples of dot-width by an integer in the main and the sub-scanning directions of the dot type printer 300. It is further provided with a function to approximate the distance thus obtained to a position either immediately before or after it of those distances in the multiples of dot-width by an integer. Approximation is carried out by rounding up, rounding off or the method using threshold values. In the cases of rounding up and off, the errors in the line intervals and character intervals and the errors between the reference point and the last character of lines and between the reference point and the last line concentrate within a range of one fold dot-width or less while in the case of rounding fractions of 5 and over to the nearest whole number and disregarding the rest, the errors in the character intervals and in the line intervals concentrate within a dot width or less and the last character of each line and the last line remain within the limit of $\frac{1}{2}$ of the dot-width from the reference point.

FIG. 2 schematically indicates the control unit 200 when it is applied to the control in the main scanning direction, comprising a character code buffer 1, a character clock system 2, a fax clock system 3, an approximation circuit 5 and a shift register 6. The character code buffer 1 retains code rows which correspond to character lines including one line equivalent blank codes which are fed from CPU 100 with the character coding signal 201*b* of the printer signal 201. The character clock system 2 is driven by a character clock which corresponds to the correct width of a character and outputs a correct position or a correct distance of each character from the home position as a character pulse CK_{60n}. The fax clock system 3 is driven by a fax clock which corresponds to the dot-width in the main scanning direction of the dot-type printer and outputs dot data 4 of characters using the dot-width of the fax in the main scanning direction as a unit. The approximation circuit 5 outputs an approximation pulse CK_{60apx} or the signal which approximate the character pulse CK_{60n} to fax clock CK_8. The shift register 6 loads the dot data from the fax clock system 3 by the approximation pulse CK_{60apx} and shift the same with the fax clock CK_8. By loading with the approximation pulse CK_{60apx} each character is corrected to remain in the range of position errors in one or $\frac{1}{2}$ of the dot-width from the home position.

The present invention will further be explained referring to an example of circuit. FIG. 3 shows an example of the circuit of the control unit 200. This shows the case where both main and sub-scanning directions are approximated by counting fractions of 5 and over as a unit and disregarding the rest.

In FIG. 3 the reference numeral 7 denotes a fax clock generator which outputs fax clock CK_8 and clock CK_8 which is delayed in phase by π than CK_8. The cycle T_8 of the clocks CK_8 and CK_8 is identical with the transmitting cycle of a picture signal 301 to the dot-type printer 300 and is assumed to have the duty factor of 50%. The reference numeral 8 denotes a character clock generator. The cycle T_{60} of the character clock CK_{60} transmitted from the generator is determined by the equation below since the character width is a multiple of 1/60 inch by an integer while the resolution in the main scanning direction is 8.04 pel/mm:

$$\frac{T_{60}}{T_8} = \frac{25.4/60}{1/8.04} \approx 3.4 \qquad \text{Formula (1)}$$

Those fax clock CK_8 and character clock CK_{60} are made synchronized to the line synchronous signal 302*b* fed from the dot-type printer 300 while synchronization

and duty factor thereof are determined by such a conventional technique as by dividing frequency of a higher harmonic. FIG. 4 indicates the relation between the line synchronous signal 302*b* and the page synchronous signal 302*a* whereas FIG. 5 shows the relation between those synchronous signals and the fax clock CK_8 and the character clock CK_{60}. The reference numeral 9 denotes a character width generator. Since in the case of proportional spacing characters the width varies depending on characters, this example uses a read-only memory which stores various character widths as a value which is smaller by [1] than an integer calculated from the unit of 1/60 inch. For example, if ASCII code is used as the character code, since the width of the letter "i" is 3/60 inch, [2] is outputted to the sexadecimal notation (69)H of ASCII code which expresses "i" whereas since the width of the letter "w" is 8/50 inch, [7] is outputted to (57)H of ASCII code.

The reference numeral 10 denotes a character generator which is a read-only memory to store the forms of various characters in the form of dot matrixes. The maximum width in this case is assumed to be at 8/60 inch, the maximum height at 1/60 inch and the size of the dot matrix at 28×32 so as to be applied to characters of 3/60 to 8/60 inch in width as shown in FIGS. 6(*a*) to (*f*). As the approximation error in character width remains equal to the dot-width or less, various character widths are approximated to either one of integers which are made by rounding up or off the fractions of the pel numbers indicated in Table 1. The width of the character frame at the character generator 10 therefore becomes an integer obtained by rounding up the pel number in Table 1. In order to secure an appropriate character width, the two bits at the extremely right within the character frame is made not to contain characters. It is also so adopted that a character of smaller width is to have a smaller frame deflected to the left and that the right side indicated by hatches in FIG. 6 is to be used as a blank space. The types of characters are determined by the character address ADC from the character code buffer 1 and while the value of the character line address ADCL remains between 0 to 31, a dot data 4 of 28 bits of the line corresponding to ADCL value out of the dot matrix of the said letter is outputted to the shift register 6. When the ADCL value is 32 or more, a dot row expressing a white or blank line is outputted.

The reference numeral 11 denotes a character width counter which counts the character clock CK_{60} corresponding to 1/60 inch so as to measure the time corresponding to the character widths of various characters. The character width counter 11 in this example receives the output from the character width generator 9 to count down the character clock CK_{60}. The bolo output CK_{60n} of the character width counter is the character pulse. The loading of the character counter 11 is carried out by the pulse signal CK_{60nd} which is obtained by triggering a monostable multi-viblator 12 with the rise of the character pulse CK_{60n}. The pulse signal CK_{60nd} used for loading is also used as a count-up input to the character code buffer address counter 13. It is cleared with the line synchronous signal 302*b* and then reads out the contents of the character code buffer 1 or in other words the character address ADC for every count-up.

The functions and performances above mentioned will now be indicated in FIGS. 7(*a*) to (*l*) for the case where the character width of the first character is 3/60 inch and that of the second character is 4/60 inch. The time of fall of the character pulse CK_{60} correctly registers with the left point of the character frame of each character as shown in the drawings.

In this embodiment, the approximation circuit 5 comprises a monostable multi-vibrator 30 and D type flip-flop 31. The preset input of the D type flip-flop 31 is fed with the line synchronous signal 302*b* via an inverter 32 whereas the clock input is fed with the exclusive "OR" of the $\overline{Q}$ outputs of the character pulse CK_{60n} and the monostable multi-vibrator 30 which are obtained from the gate 33. The data input of the D type flip-flop 31 is fed with its own $\overline{Q}$ output and is provided with the function of a T type flip-flop. The monostable multivibrator 30 is fed with the $\overline{Q}$ output CK_{60np} of the D type flip-flop 31 for clear input and simultaneously with the fax clock CK_8 having the duty factor of 50% for trigger input in order to output a negative pulse. This negative pulse is the approximation pulse CK_{60apx} rounded by the method of counting fractions of 5 and over as one unit and disregarding the rest. As indicated in FIGS. 7(*b*), (*d*), (*e*) and (*i*), one negative pulse CK_{60apx} is outputted at the rise of the fax clock CK_8 immediately after the fall of the character pulse CK_{60n}. If it is assumed that the time difference in the falls between CK_{60n} and CK_{60apx} is E_{60n} at the time of the fall of CK_{60n}, the following formulae hold:

(1) If CK_8 is at a high level, $\frac{1}{2}T_8 < E_{60n} < T_8$ . . . Formula (2)

(2) If CK_8 is at a low level, $0 < E_{60n} \leqq \frac{1}{2}T_8$ . . . Formula (3)

Since the time error at the top of a line is always fixed at $\frac{1}{2}T_8$, the value Eh or the value obtained by substracting the said error $\frac{1}{2}T_8$ from all E_{60n} can be obtained as

Formula (4)

$$\left.\begin{array}{l}\text{in the case of (2)} \ldots 0 < Eh < \frac{1}{2}T_8 \\ \text{in the case of (3)} \ldots -\frac{1}{2}T_8 < Eh \leqq 0\end{array}\right\}$$

This means CK_{60apx} approximates CK_{60n} to the unit of CK_8 by counting the fractions of 5 and over as one unit and disregarding the rest. Compared to the character pulse of the correct time CK_{60n}, the time of the approximation pulse CK_{60apx} from the home position remains in the scope of $\pm\frac{1}{2}T_8$, or in other words, within the error of one half of the dot-width in the main scanning direction. The error between adjacent approximated pulses is within the scope of $\pm T_8$ or, in other words, one-fold dot width or less.

The shift register 6 is a shift register with 29 bit parallel input wherein an approximation pulse CK_{60apx} is naturally used for load input of 28-bit dot data and fax clock CK_8 is used for shift clock. The output Vdf of the shift register 6 is delayed in the synchronization by π from the line synchronous signal 302*b*. In order to coincide the phase thereof by further delaying it by π, the output Vdf from the shift register 6 is supplied to the D input of the phase regulator of the D type flip-flop, and the clock CK_8 which is delayed by π from the fax clock CK_8 is fed as a clock input, thereby obtaining a picture signal 301. By conducting such a phase regulation, the whole errors between characters and lines will be made concentrated within the scope mentioned above. As the first picture element of the line becomes white, no ill effect is produced.

As described in the foregoings, every time an approximation pulse CK_{60apx} is generated, the dot data from the character generator 10 is loaded in the shift register

6 and then is shifted by the fax clock CK_8, thereby keeping the positions of respective characters from the home position within the error of one half of the dot-width and keeping the error between characters within one-fold thereof. In FIGS. 7(*j*) and (*l*) the symbol B_{ij} denotes the j-th dot data at the i-th character.

The above explains about the main scanning direction but the same is applicable to the sub-scanning direction, which will now be described referring to the waveforms of operation shown in FIG. 8. In FIG. 3, the reference numeral 15 denotes a line clock generator which outputs a line clock CK_7 and a clock CK_7 which is delayed by π therefrom. Those clocks CK_7 and CK_7 are synchronized with the line synchronous signal 302*b* from the dot type printer 300 with the cycle T_7 identical to the line synchronous signal 302*b* and of the duty factor of 50%. The reference numeral 16 denotes a line clock generator. The line clock CK_{12} generated therefrom is synchronized with the page synchronous signal 302*a*. As the resolution in the sub-scanning direction is 7.7 whilst the line width is a multiple of 1/12 inch by an integer, the cycle T_{12} can be determined by the formula (5) below:

$$\frac{T_{12}}{T_7} = \frac{25.4/12}{1/7.7} \approx 16.30 \qquad \text{Formula (5)}$$

The reference numeral 17 denotes a line width register which receives a line width signal 201*a* contained in the printer signal 201 from CPU 100 to memorize a value which is smaller by [1] than an integer expressed by using the unit of 1/12 inch.

The reference numeral 18 denotes a line width counter which measures the time corresponding to the correct line width for each line by counting the line clock CK_{12} which corresponds to 1/12. The line width counter 18 operates in a manner similar to the aforementioned character width counter 11 to load the content of the line width register 17 and to output a bolo output CK_{12n} by counting down the line clock CK_{12}. It is loaded with the pulse CK_{12nd} which is obtained by triggering the monostable multivibrator 19 at the rise of the line pulse CK_{12n}.

The reference numeral 20 denotes an approximation circuit which in this embodiment comprises D type flip-flops 21 and 35 and a monostable multivibrator 22. The preset input of the D type flip-flop 21 is fed with a page synchronous signal 302*a* via an inverter 23 while the clock input thereof is fed with a line clock CK_7, and the data input with a pulse CK_{12}. The preset input of the D type flip-flop 35 is fed with a page synchronous signal 302*a* via an inverter 23 while the clock input thereof is fed with exclusive "OR" of Q output CK_{12nq} of the D type flip-flop 21 and $\overline{Q}$ output of the monostable multivibrator 22 obtained from the gate 36. It further supplies the data input thereof to its own $\overline{Q}$ output so as to give the function of T type flip-flop thereto. The Q output from the D type flip-flop 35 is supplied to the monostable multi-vibrator 22 at the clear input thereof and simultaneously the clock CK_7 of duty factor of 50% is given thereto as a rise trigger input, thereby obtaining a rounded positive pulse or the approximated line pulse CK_{12apx}. In short, as shown in FIGS. 8(*c*) to (*h*), one positive pulse CK_{12apx} is outputted at the rise of the clock CK_7 following the rise of the line clock CK_7 immediately after the fall of the line pulse CK_{12n} or in other words, at the fall of the line clock CK_7 immediately after the fall of the line pulse CK_{12n}. Assuming the time difference between the falls of CK_{12n} and corresponding CK_{12apx} is E_{12n}, the following formulae hold:

(a) If CK_7 is at a high level at the fall of Ck_{12n}, $T_7 < E_{12n} < 3/2T_7$ . . . Formula (6)

(b) If CK_7 is at a low level at the fall of CK_{12n}, $\frac{1}{2}T_7 < E_{12n} \leqq T_7$ . . . Formula (7)

Since E_{12n} at the top of a page is always T_7, the value obtained by substracting the initial error T_7 from all the E_{12n}s becomes

$$\left.\begin{array}{l}\text{in the case of (6) } 0 < Ev < \frac{1}{2}T_7 \\ \text{in the case of (7) } -\frac{1}{2}T_7 < Ev \leq 0\end{array}\right\} \qquad \text{Formula (8)}$$

That means CK_{12apx} is approximated rounding CK_{12n} to the unit of CK_7 by the method of counting the fractions of 5 and over as a unit and disregarding the rest. By this manipulation, the time of the approximation pulse CK_{12apx} from the top of a page or from the home position concentrates within the scope of error of $\pm\frac{1}{2}T_7$ or one half of the dot width in the sub-scanning direction (T_7).

The reference numeral 24 denotes a line counter which receives the approximation pulse CK_{12apx} of a line width from the approximation circuit 20 and a page synchronous signal 302*a* via an OR gate 25 as a clear input and receives clock CK_7 which is delayed by π from the line clock CK_7 as a count-up input. (Refer to FIG. 8 (*i*)). The output from this line counter 24 is the character line address ADCL to be supplied to the character generator 10. By clearing the character line address ADCL with the approximation pulse CK_{12apx}, positions of each line from the page top concentrate within the one half of the dot-width error while the line widths concentrate within the scope of dot width with the error of one-fold thereof. The content of the line counter 24 will be delayed by one line if it is simply cleared with an approximation pulse CK_{12apx} although it is synchronized in phase with the page synchronous signal 302*a*. As shown in the figure, it therefore is given with a page synchronous signal 302*a* in the clear input in order to cause the content of the line counter 24 to become [0] for the first one line. Since the first line becomes white by giving a blank code to the character code buffer 1 as for the first row, there remain no adverse effects.

The content of the line counter 24 also doubles as the control signal 202 to CPU 100 from which CPU 100 can recognize the end of printing of each line so as to supply the signal 201*b* of the character code row corresponding to the following line to the character code buffer 1 and simultaneously to supply a line width to the line width register 17.

The embodiment described in the foregoings with the reference made to FIGS. 1 to 8 relates to the case wherein the system of generating character codes exists at the same place as the dot type printer. FIG. 9, on the other hand, shows another system wherein those two places exist separately at remote and separate places and character codes are converted to facsimile data on the side of transmitter to be transmitted over a communication circuit to a dot type printer.

In FIG. 9 the reference numeral 100 denotes a CPU, 200 a control system according to the present invention, 300 a dot type printer, 400 a communication control unit, and 500 a communication circuit. The reference numeral 201 denotes printer signals from the CPU 100

to the control unit 200, including character code row signals expressing the character rows of each line, and signals expressing the line intervals. The reference numeral 202 denotes a signal which expresses the end of lines and which is sent from the control unit 202 to the CPU 100. In this embodiment the signals 302 of either page synchronous or of line synchronous from the control unit 200 are fed from the communication control unit 400 and the picture signal 301 from the control unit 200 is transmitted to the dot type printer 300 with the appropriate timing via the communication control unit 400.

In the case of the system wherein character data in a manuscript is transmitted in the form of a character code and drawing data in a form of a facsimile code, if the part to be converted into the character codes is converted into facsimile codes by using the output control system according to the present invention, picture signals with minimum errors in the positions of character can be inputted into the dot type printer.

I claim:

1. An output control system for dot type printers having a fixed resolution in either main scanning direction or sub-scanning direction which is characterized in that when an output object of a pitch which does not coincide with multiples of the dot width by integers, the distance from a reference point in the direction with the fixed resolution is obtained for respective objects, and that said obtained distance is approximated to a distance immediately before or after such distance selected out of those separated from the said reference point by multiples of the dot width by integers.

2. The output control system for dot type printers as claimed in claim 1 which is characterized in that at the time of approximation of the said objects, the distance of the objects from the reference point is calculated by using the unit of the dot width, and that such value of the distance is approximated by rounding off the value to an integer immediately below if the fraction decimal order is smaller than a predetermined threshold value and by rounding up the value to an integer immediately above if the decimal fraction is larger than the threshold value.

3. The output control system for dot type printers as claimed in claim 2 which is characterized in that the said threshold value is 0.5.

* * * * *

United States Patent [19]

Gojo

[11] **Patent Number: 4,555,191**

[45] **Date of Patent: Nov. 26, 1985**

[54] **METHOD OF REDUCING CHARACTER FONT**

[75] Inventor: **Tomomi Gojo, Zama, Japan**

[73] Assignee: **Ricoh Company, Ltd., Tokyo, Japan**

[21] Appl. No.: **668,319**

[22] Filed: **Nov. 5, 1984**

[30] **Foreign Application Priority Data**

Nov. 5, 1983 [JP] Japan 58-206914

[51] **Int. Cl.**[4] **B41J 3/02; H04N 1/00**

[52] **U.S. Cl.** **400/121;** 340/728; 340/731; 340/735; 340/745; 382/56; 400/110

[58] **Field of Search** 400/121, 110, 62, 124; 340/728, 731, 735, 745; 358/263; 382/56

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,604,846	9/1971	Behane	358/263
4,090,188	5/1978	Suga	340/731
4,129,860	12/1978	Yonezawa	340/728
4,216,480	8/1980	Buehner	400/126 X
4,242,678	12/1980	Somerville	340/728
4,288,816	9/1981	Kashioka et al.	382/56
4,476,464	10/1984	Hobbs	340/731

FOREIGN PATENT DOCUMENTS

8024	1/1978	Japan	340/731
14674	1/1983	Japan	358/263

Primary Examiner—Paul T. Sewell
Attorney, Agent, or Firm—Oblon, Fisher, Spivak, McClelland & Maier

[57] **ABSTRACT**

A method of reducing the size of a character font in a word processor or an office computer. Digitized font data representative of a character font is constructed in a dot matrix consisting of a plurality of rows in an X direction and a plurality of columns in a Y direction perpendicular to the X direction. The first row of dots in the X direction is divided into a plurality of blocks and each block is provided with a ONE or a ZERO depending upon the number of dots included therein. The first row is shifted one dot to the left in the X direction to form a shifted row. The shifted row is divided into a plurality of blocks and each of these blocks is provided with a ONE or a ZERO depending upon the number of dots included therein. ANDs of the respective logical values of the shifted and non-shifted rows in the Y direction are computed to obtain a ONE or ZERO block by block. These consecutive steps are repeated on all the rows in the X direction to thereby provide a ONE or a ZERO for each of the blocks in all the rows. Each of the columns in the Y direction represented by ONEs and ZEROs of the blocks in the X direction is divided into a plurality of blocks. ANDs in the Y direction are computed block by block to decide the blocks whose ANDs are ONEs to be black dots.

5 Claims, 17 Drawing Figures

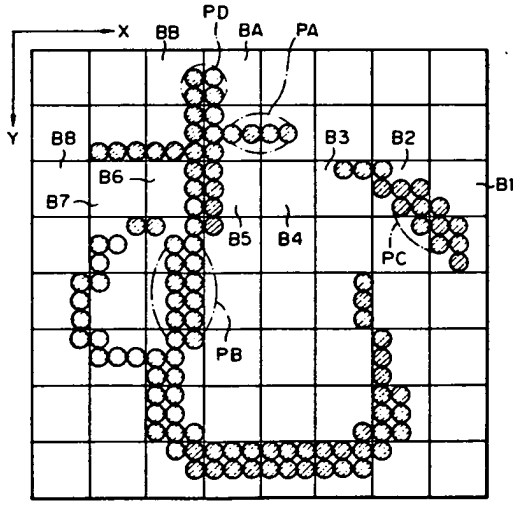


Fig. 1

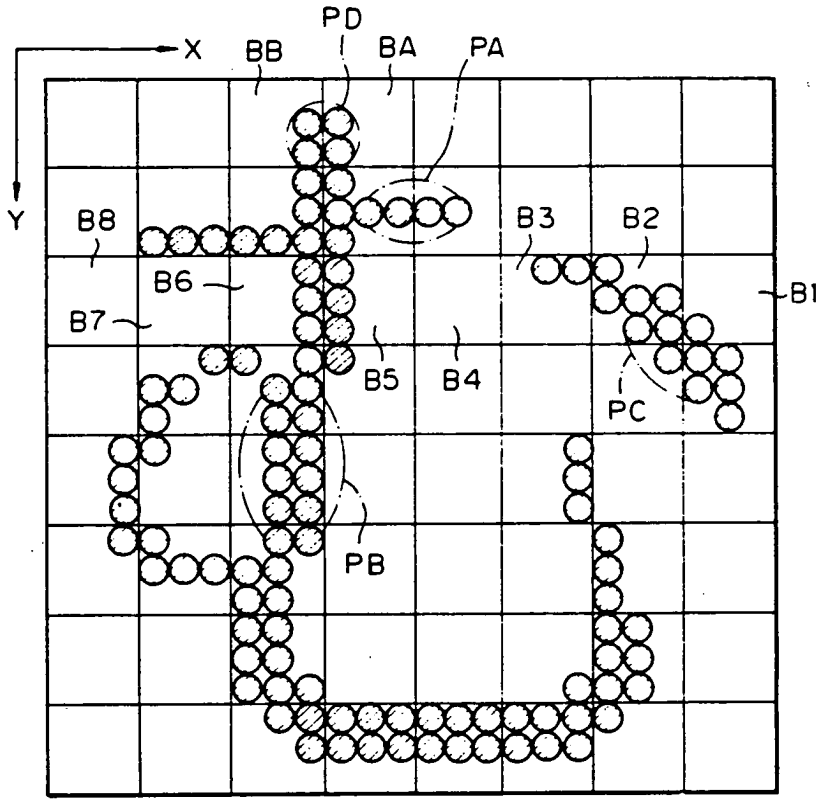


Fig. 2

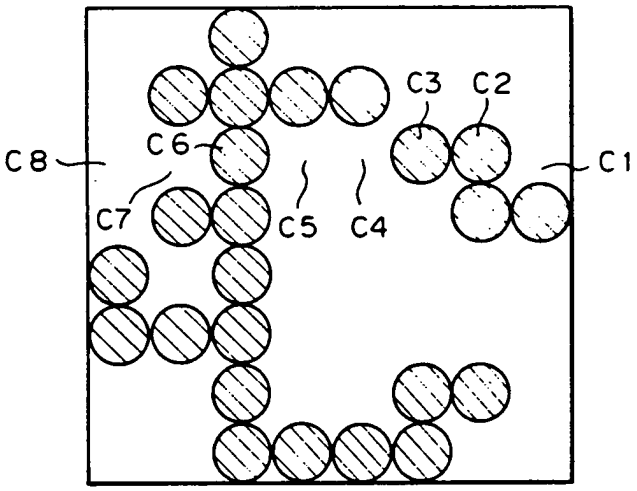


Fig. 3A

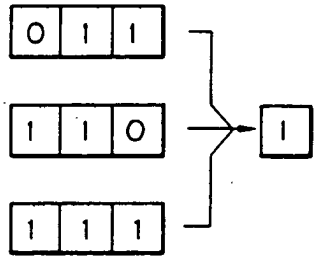


Fig. 3B

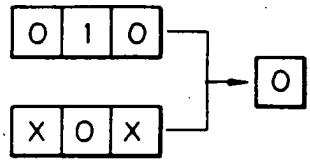


Fig. 4A

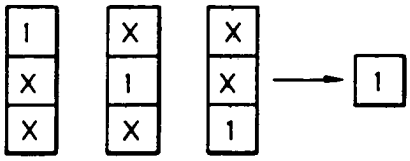


Fig. 4B

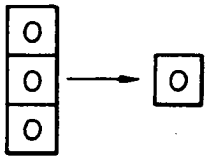


Fig. 5A

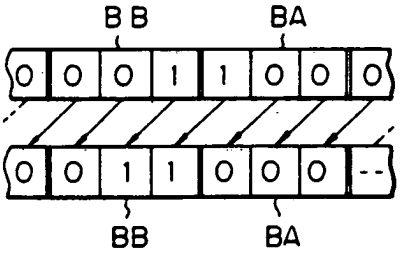


Fig. 5B

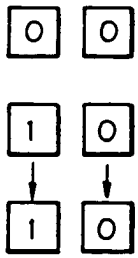


Fig. 6

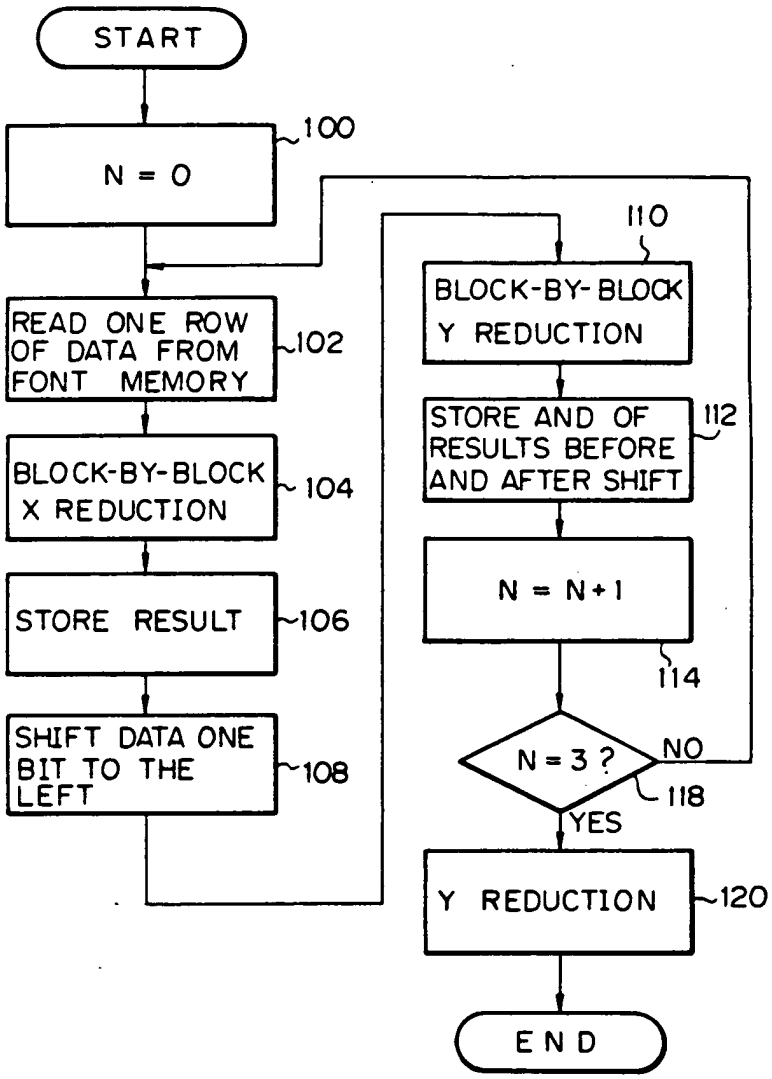

START
N = 0
100
READ ONE ROW OF DATA FROM FONT MEMORY
102
BLOCK-BY-BLOCK X REDUCTION
104
STORE RESULT
106
SHIFT DATA ONE BIT TO THE LEFT
108
110
BLOCK-BY-BLOCK Y REDUCTION
STORE AND OF RESULTS BEFORE AND AFTER SHIFT
112
N = N+1
114
N = 3 ?
NO
118
YES
Y REDUCTION
120
END

Fig. 7

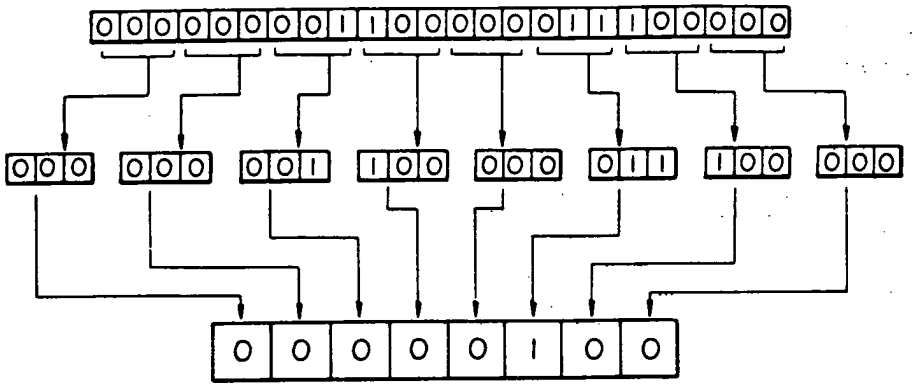


Fig. 8A

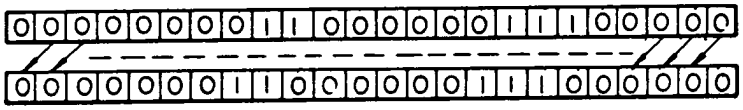


Fig. 8B

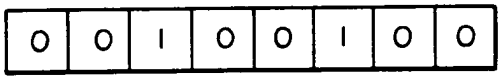


Fig. 9

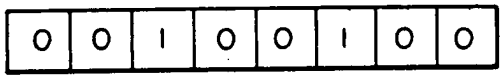
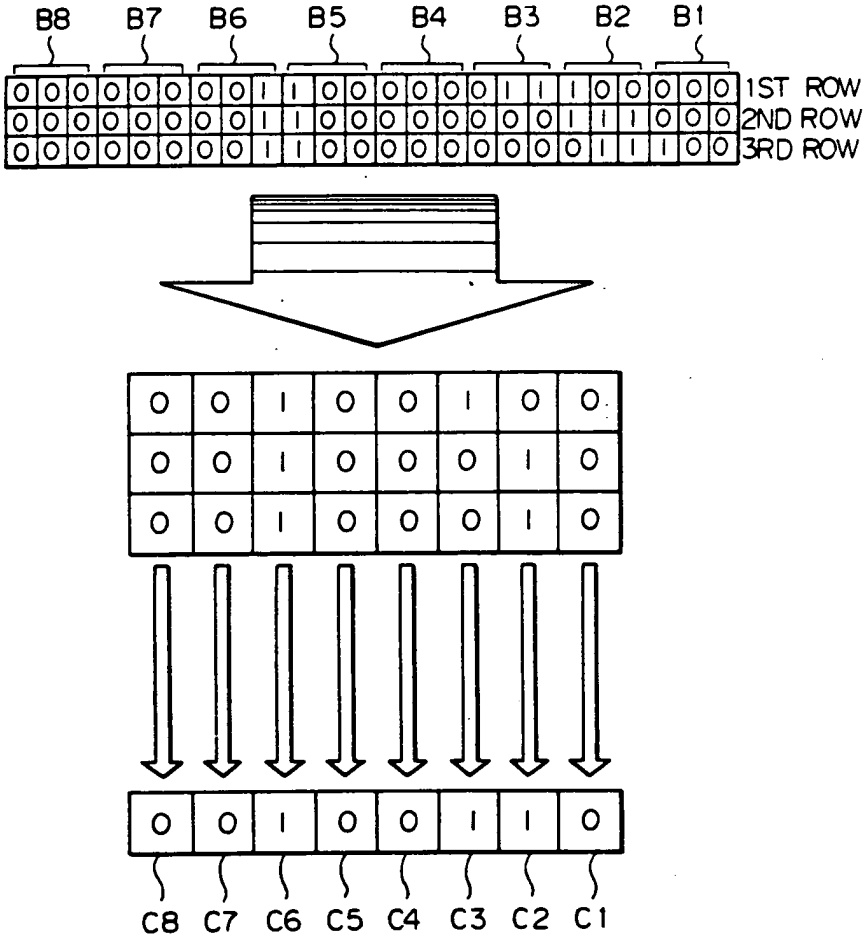


Fig. 10


B8 B7 B6 B5 B4 B3 B2 B1
1ST ROW
2ND ROW
3RD ROW
C8 C7 C6 C5 C4 C3 C2 C1

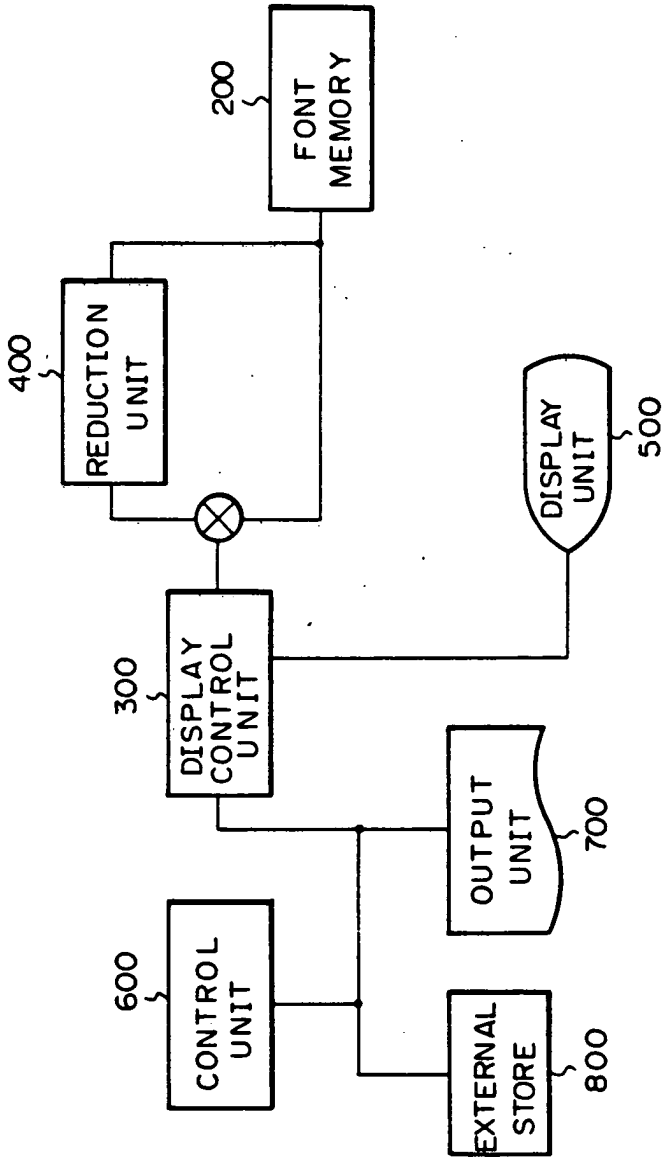

200
FONT MEMORY
400
REDUCTION UNIT
DISPLAY UNIT
500
300
DISPLAY CONTROL UNIT
OUTPUT UNIT
700
600
CONTROL UNIT
EXTERNAL STORE
800

Fig. 11

Fig. 12

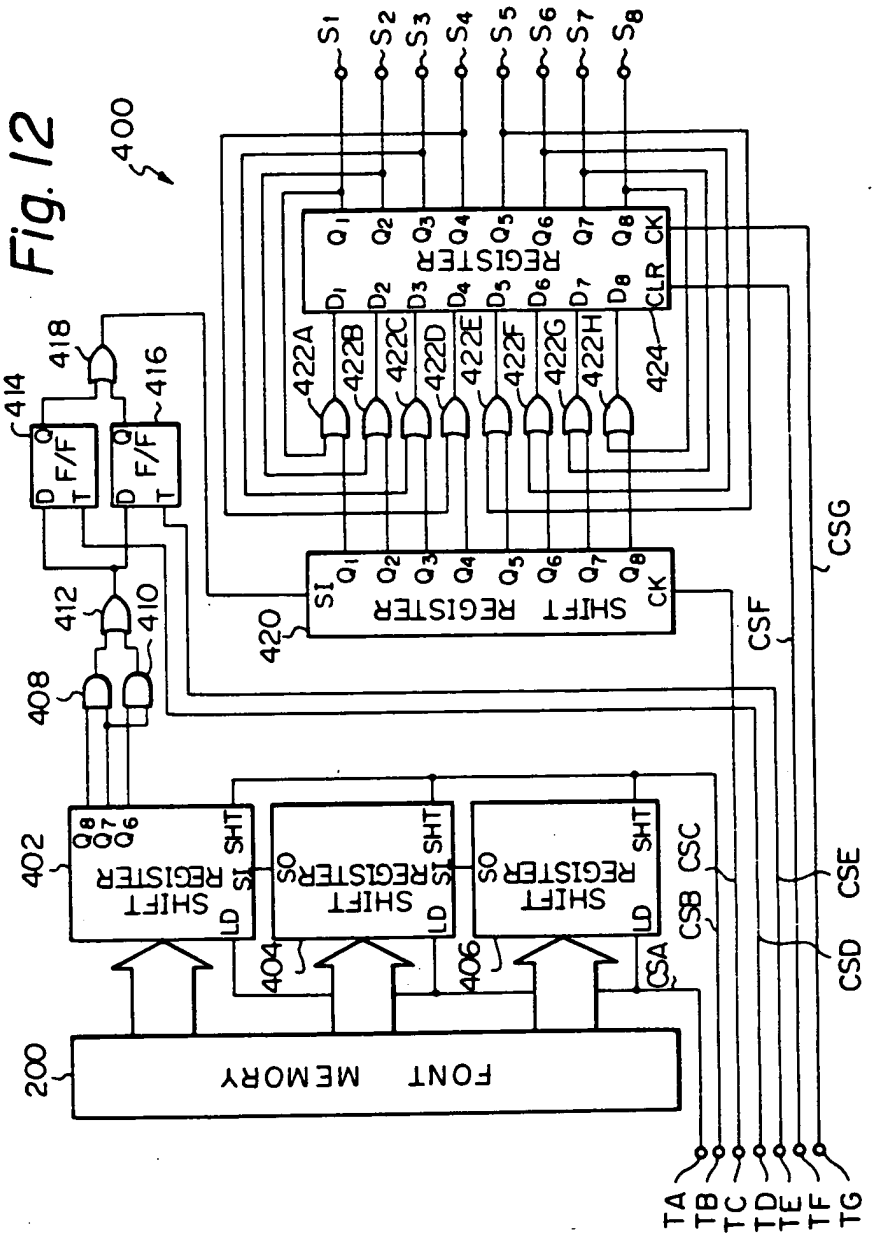
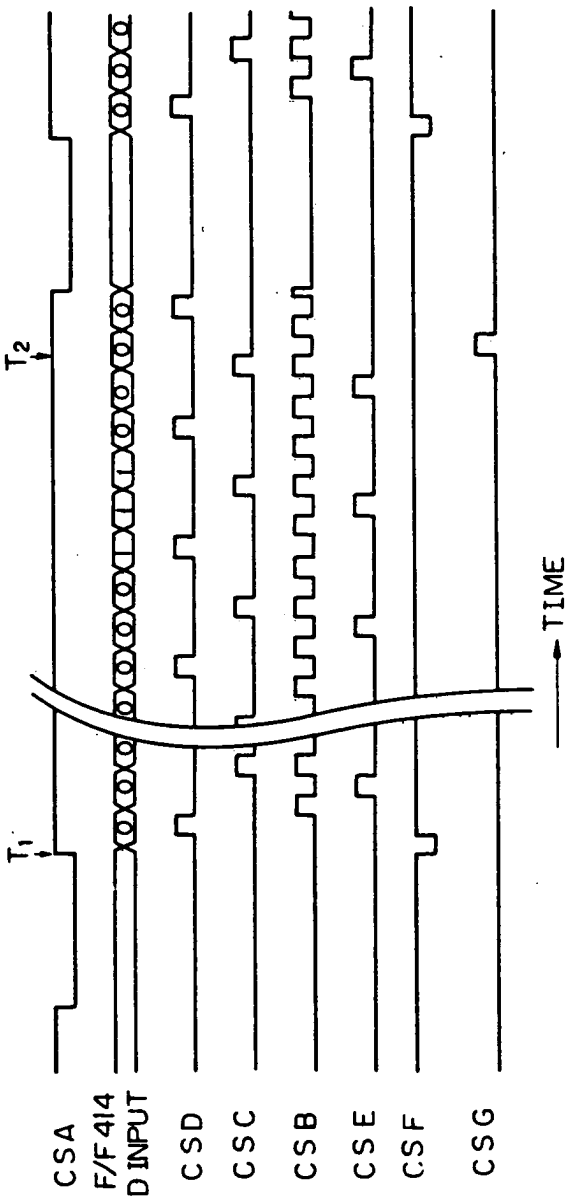

400
S1
S2
S3
S4
S5
S6
S7
S8
REGISTER
Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 CK
D1 D2 D3 D4 D5 D6 D7 D8 CLR
422A
422B
422C
422D
422E
422F
422G
422H
424
418
416
414
D F/F Q
T
412
410
408
SHIFT REGISTER
SI Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 CK
420
CSF
CSG
402
SHIFT REGISTER
Q8 Q7 Q6 SI SHT LD
404
406
CSA
CSB
CSC
CSD
CSE
200
FONT MEMORY
TA
TB
TC
TD
TE
TF
TG

Fig. 13



METHOD OF REDUCING CHARACTER FONT

BACKGROUND OF THE INVENTION

The present invention relates to a method of varying the font size of characters, numerals, marks, symbols and the like (hereinafter generally referred to as "characters" for convenience) and, more particularly, to a method of reducing a character font size.

Word processors, office computers and the like are modern implementations extensively used for documents processing. In such implementations, besides ordinary characters consisting of a 24 by 25 dot matrix, characters in a smaller size or a larger size are often required. Especially, in Japanese, the demand for reduction of the character size is keen due to double constants and contracted sound involved therein. Concerning Japense, even a word processor or the like has come to be required to prepare a harmonious and orderly document as experienced in showing the reading of addresses and names by printing kana at their sides and such has been reflected by an increasing demand for reduced character fonts.

One approach to realize document processing with a reduced character size is storing the fonts of reduced characters in advance in a font memory. This, however, would require a font memory having an increased capacity. Another approach is generating reduced fonts by means of a word processor or like apparatus. For example, in a layout mode of a word processor designed for Japanese, exclusive patterns for discriminating kanji (Chinese characters) and the other characters is available for selecting a layout in a document page. The problem with the predetermined pattern scheme is that although one may readily grasp the layout in one document page, he or she experiences difficulty in recognizing a positional relationship between sentences and graphs. While such problems may be settled if the operator himself prepares reduced documents, the operation would consume disproportionate time and labor.

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a method which readily achieves a reduced character font without inviting any increase in the required capacity of a font memory.

It is another object of the present invention to provide a generally improved method of reducing a character font.

A method of reducing a character font in which font data representative of the character font consists of a dot matrix having a plurality of rows of arranged in a first direction and a plurality of columns of arranged in a second direction perpendicular to the first direction of the present invention comprises the steps of (a) producing first data by dividing one of the rows in the first direction into a plurality of blocks such that each of the blocks includes a predetermined number of dots and, then, converting the one row to unit dots associated with the respective blocks, (b) producing second data by shifting the row in the step (a) in the first direction to constitute a shifted row, then dividing the shifted row into a plurality of blocks, and then converting the shifted row to unit dots associated with the respective blocks, (c) computing third data by associating the first and second data with each other in the second direction, (d) computing third, row-by-row data by sequentially performing the steps (a), (b) and (c) in the second direction on the rows of a number which corresponds to the predetermined number, (e) computing fourth data by associating in the second direction the third, row-by-row data computed by the step (d), (f) computing the fourth data associated with the whole character font by repeating the consecutive steps (a) through (e), and (g) converting the respective blocks consisting of dots associated with the plurality of fourth data computed by the step (f) to unit dots associated with the respective blocks.

In accordance with the present invention, a method of reducing the size of a character font in a word processor or an office computer is provided. Digitized font data representative of a character font is constructed in a dot matrix consisting of a plurality of rows in an X direction and a plurality of columns in a Y direction perpendicular to the X direction. The first row of dots in the X direction is divided into a plurality of blocks and each block is provided with a ONE or a ZERO depending upon the number of dots included therein. The first row is shifted one dot to the left in the X direction to form a shifted row. The shifted row is divided into a plurality of blocks and each of these blocks is provided with a ONE or a ZERO depending upon the number of dots included therein. ANDs of the respective logical values of the shifted and non-shifted rows in the Y direction are computed to obtain a ONE or ZERO block by block. These consecutive steps are repeated on all the rows in the X direction to thereby provide a ONE or a ZERO for each of the blocks in all the rows. Each of the columns in the Y direction represented by ONEs and ZEROs of the blocks in the X direction is divided into a plurality of blocks. ANDs in the Y direction are computed block by block to decide the blocks whose ANDs are ONEs to be black dots.

The above and other objects, features and advantages of the present invention will become more apparent from the following detailed description taken with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows an example of a character font which is represented by a 24 by 24 dot matrix;

FIG. 2 shows an example of a font which is reduced in accordance with the present invention;

FIGS. 3A and 3B show the principle of reduction in an X direction in accordance with the present invention;

FIGS. 4A and 4B show the principle of reduction in a Y direction in accordance with the present invention;

FIGS. 5A and 5B show the principle of the X direction reduction which involves shifting of data;

FIG. 6 is a flowchart demonstrating an example of the whole procedure for attaining a reduced font in accordance with the present invention;

FIG. 7 shows an exemplary operation for reduction in the X direction;

FIGS. 8A and 8B show an exemplary operation for reduction in the X direction with shifting involved;

FIG. 9 shows an example of the result of reduction in the X direction; and

FIG. 10 shows an example of the overall reduction operations in the X and Y directions;

FIG. 11 is a block diagram of an exemplary system to which a character font reduction device in accordance with the present invention is applicable;

FIG. 12 is a block diagram showing an embodiment of the character font reduction device in accordance with the present invention; and

FIG. 13 is a timing chart representative of operation of the device shown in FIG. 12.

DESCRIPTION OF THE PREFERRED EMBODIMENT

While the method of reducing character font of the present invention is susceptible of numerous physical embodiments, depending upon the environment and requirements of use, a substantial number of the herein shown and described embodiments have been made, tested and used, and all have performed in an eminently satisfactory manner.

The construction in accordance with the present invention will be described in conjunction with one embodment thereof. In this particular embodiment, reduction of a 24 by 24 dot matrix to an 8 by 8 dot matrix is assumed by way of example.

First, the principles of font reduction in accordance with the present invention will be described. In FIG. 1, hiragana (cursive kana character) " " is shown as an example of 24 by 24 dot characters. When this front is reduced to an 8 by 8 dot font, it appears as shown in FIG. 2. The font shown in FIG. 1 is reduced on the basis of a unit block which consists of a 3 by 3 dot submatrix.

Referring to FIGS. 3–5, a specific conversion method for font reduction is shown. The conversion is attained paying attention to characteristics particular to a font which will be described. Regarding the font shown in FIG. 1, the dot density is higher in the Y or vertical direction than in the X or horizontal direction. That is, while a single dot-string arranement is predominant in the X direction strokes (see PA in FIG. 1), a two dot-strings arrangement is predominant in the Y direction strokes (see PB and PC in FIG. 1). Imaging other kinds of characters will make such tendency more clear. Stated another way, the tendency described above is generally observed in kana, kanji and like characters.

In light of the above characteristics, the reduction in the X direction, or simply X reduction as will be referred to, is performed such that when nearby two bits out of the three bits in each unit block are (binary) "1", meaning that the nearly two dots should be printed out as black dots, the entire block is reduced as "1" (see FIG. 3A). When otherwise, the block is reduced as "0", meaning that it will not be printed out as a dot (see FIG, 3B). In the drawings, the mark "x" shows that bits so marked are either "1" or "0" (DONT'T CARE).

Concerning the reduction in the Y direction, or Y reduction, if at least one bit in the three-bit unit block is "1", the whole block is decided to be "1" and, if otherwise, "0" (see FIGS. 4A and 4B). That is, the reduction effected by performing an AND operation. By the above procedure, the font represented by the 3 by 3 bit block is first reduced by the X reduction to a 3 by 1 bit block and, then, to a 1 by 1 bit font by the Y direction. The sequence of the X reduction and the Y reduction described is not limitative and may be inverted.

The font reduction relying only on the procedure described above entails the following inconvenience. When the X reduction is performed by the above procedure on such a portion as PD shown in FIG. 1 where the Y direction stroke extends over two nearby blocks, both the nearby blocks will be undesirably converted to "0".

The present invention overcomes such inconvenience by employing the following extra operation together with the X reduction shown in FIGS. 3A and 3B. Specifically, the principles of reduction shown in FIGS. 3A and 3B are applied to the stream of bits which are shifted one bit to the left as shown in FIG. 5A and, then, there is provided ANDs of the result of such reduction and the result of reduction shown in FIGS. 3A and 3B which is applied to a state before the shift (see FIG. 5B). It will be noted that the leftward shift, not the rightward, is employed in view of the fact that a 24 by 24 bit font generally carries a far smaller amount of information in the leftmost column than in the rightmost column.

Spcifically, in FIG. 1, observing the second row in nearby blocks BA and BB which include the portion PD, the block BA before the shift is "100" and the block BB "001" and, hence, both of them become "0" after the reduction shown in FIG. 3B (see the top of FIG. 5B). After the shift, the block BA turns to "000" and the block BB to "011" as shown in FIG. 5A. Applying reduction to such bits of the blocks BA and BB makes the block BA "0" as shown in FIG. 3B and the block BB "1" as shown in FIG. 3A (see the middle of FIG. 5B). After the AND operation, the block BA becomes "0" and the block BB "1" and this is the result of the X reduction (see the bottom of FIG. 5B).

Referring to FIG, 6, an exemplary procedure is shown which may be utilized with a computer or the like for the above-described font reduction. The flowchart shown in FIG. 6 represents a procedure associated with one row of blocks extending in the X direction. In the case where the 24 by 24 bit font shown in FIG. 1 is to be reduced to the 8 by 8 bit font, the flow shown in FIG. 6 is repeated eight consecutive times. The reduction procedure will be discussed taking blocks B1 to B8 of FIG. 1 for example.

The procedure starts with setting up a condition "N=0" (100). Here, "N" represents a counter responsive to the number of times of reduction occurring in the X direction and is necessary for the X reduction to be performed on three lines, the counter being initially set to "0".

Next, one row of font data are read out of a font memory as shown in the top of FIG. 7 (102). The font data are divided into each three bits (see the middle of FIG. 7) and, then, the X reduction is performed as described in conjunction with FIGS. 3A and 3B (104) (see the middle of FIG. 7). The result is stored in a suitable storage (106) (see the bottom of FIG. 7).

Then, the font data are bodily shifted one bit to the left as shown in FIG. 8A (108) whereafter the X reduction occurs block by block (110) (see FIG. 7). The result is shown in FIG. 8B. This result and the result of reduction before the shift shown in FIG. 7 are processed to provide their AND which is then stored in a storage (112). The resulting row of eight bits is shown in FIG. 9.

By the operation described so far, the X reduction on the first row of data in the blocks B1 to B8 is completed. Then, the counter is incremented to N=N+1 (114).

The above procedure is repeated on each of the second and third rows of font data (118), the results being shown in the middle of FIG. 10. Shown in the top of FIG. 10 is the font of the blocks B1 to B8 of FIG. 1 which are represented by "1" and "0".

Thereafter, Y reduction is performed (120) on the data which underwent the X reduction as described.

Data provided by the Y reduction are shown in the bottom of FIG. 10.

By the procedure discussed above, the font data in the blocks B1 to B8 are reduced as represented by bits C1 to C8 in FIG. 2.

With the flowchart shown in FIG. 6, a computer or like implement may be manipulated for font reduction based on software processing.

Hereinafter will be described an embodiment of a character font reduction device which is capable of effecting the font reduction based on the above-described technique.

Referring to FIG. 11, an exemplary system to which the font reduction device is applicable is shown. A font memory 200 stores fonts each in a 24 by 24 bit matrix. The font memory 200 is connected directly to a display control unit 300 and to a character font reduction unit, or device, 400, which in turn is connected to the display control unit 300. Also connected to the dislay control unit 300 are a display unit 500, a control unit 600, an output unit 700 and an external storage 800.

In the illustrated system, document data to be applied to the display unit 500 or the output unit 700 are stored in the external storage 800. The document data are transferred from the external storage 800 to the display control 300 under the control of the controller 600 and, there, transformed into font data representative of associated characters. Specifically, font data in the font memory 200 are fetched character by character to the display control 300 either directly or by way of the reduction unit 400 for font reduction, based on the document data. The display control 300, based on the fetched font data, applies the usual character patterns or the reduced character patterns to the display 500 or the output unit 700 as a document. In this manner, the reduction unit 400 may be connected to the font memory 200 so as to reduce input font data which are in a usual size. The display control 300 actuates the output unit 700 to print out or display reduced characters in response to the output of the font reduction unit 400.

The reduction unit 400 may have such a construction as one shown in FIG. 12. In the drawing, the font memory 200 stores font data in which the font of one character is represented by 24 by 24 bits. The font data are delivered by each 24 bits and transferred to 8-bit, parallel-in serial-out type shift registers 402, 404 and 406 eight bits each.

Load terminals LD of the shift registers 402, 404 and 406 are commonly connected to a terminal TA to which a control signal CSA is applied. The control signal CSA corresponds to a read signal associated with the font memory 200 so that the font data are latched in the shift registers 402, 404 and 406 timed to the positive going edges of the control signal CSA.

Shift clock terminals SHT of the shift registers 402, 404 and 406 are connected to a terminal TB to which a control signal CSB is applied. In response to this signal CSB, the font data latched in the shift registers 402, 404 and 406 are sequentially shifted. A data output terminal SO of the shift register 406 is connected to a data input terminal SI of the shift register 406 while a data output terminal SO of the shift register 404 is connected to an input terminal SI of the shift register 402.

Output terminals Q8 and Q7 of the shift register 402 are connected to an AND gate 408 and output terminals Q7 and Q6 to an AND gate 410. Output terminals of the AND gates 408 and 410 are connected to an OR gate 412. The AND gates 408 and 410 and the OR gate 412 serves as a circuit for the X reduction which is shown in FIG. 3.

An output terminal of the OR gate 412 is connected to input terminals D of flip-flops 414 and 416. Toggle terminals of the flip-flops 414 and 416 respectively are connected to terminals TD and TE which are adapted to receive control signals CSD and CSE, respectively. Timed to the control signal CSD and CSE, the flip-flops 414 and 416 fetch data supplied to their input terminals D and apply them to the output terminals Q. The parallel connection of the flip-flops 414 and 416 is associated with the AND operation on the reduced data before the left shift and those after the left shift as shown in FIGS. 5A and 5B. The AND operation is performed by an OR gate 418 with which the output terminals Q of the flip-flops 414 and 416 are connected.

An output terminal of the OR gate 418 is connected to a data input terminal SI of a serial-in parallel-out type shift register 420. The shift register 420 has a shift clock terminal CK connecting to a terminal TC to which a control signal CSC is applied. Timed to the control signal CSC, data are stepped from the terminal IS and thereby sequentially shifted.

The shift register 420 has output terminals Q1 to Q8 which are connected to OR gates 422A to 422H, respectively. Output terminals of the OR gates 422A to 422H respectively are connected to input terminals D1 to D8 of an eight-bit register 424. Output terminals Q1 to Q8 respectively are connected to output terminals S1 to S8 of the apparatus and, also, to the other inputs of the OR gates 422A to 422H. Meanwhile, a clear terminal CLR of the register 424 is connected to a terminal TF to which a control signal CSF is applied. The control signal CSF is adapted to clear the register 424. A terminal CK of the register 424 is connected to a terminal TG to which a control signal CSG is applied. Timed to the control signal CSG, the register 424 latches and outputs data. The OR gates 422A to 422H and the register 424 constitutes as a network for the Y reduction shown in FIG. 4.

The control signals CSA to CSG applied respectively to the terminals TA to TG as described are supplied from the display control 300 or the controller 600. The output terminals S1 to S8 are connected to the display control 300.

Description will proceed based on the general operation of the embodiment described above and with reference made to the timing chart of FIG. 13 as well. The following discussion will concentrate to font reduction applied to the blocks B1 to B8 shown in FIG. 1 by way of example.

At a time T1 when the control signal CSA rises, font data representative of the first row of the blocks B1 to B8 (see top of FIG. 7 or FIG. 10) are latched in the shift registers 402, 404 and 406. Specifically, the font data in the blocks B1 to B2 and part of the blocks B3 are latched in the shift register 406; font data in the remaining part of the block B3, blocks B4 and B5 and part of the block B6 are latched in the shift register 404; and font data in the remaining part of the block B6 and in the blocks B7 and B8 are latched in the shift register 402. Meanwhile, the control signal CSF clears the register 424.

In the above condition, the shift register 402 applies to its output terminals Q8, Q7 and Q6 the font data which are shown in the leftmost position in the middle of FIG. 7. In this example, the output of the OR gate 412 associated with the leftmost data "000", i.e., the

result of X reduction shown in FIGS. 3A to 3B is "0". This is latched in the flip-flop 414 timed to the control signal CSD.

Then, timed to the control signal CSB, the contents of the shift registers 402, 404 and 406 are bodily shifted as shown in FIG. 8A. As a result, the output of the OR gate 412 turns out to be the result of the X reduction applied to the font data which underwent the shift. The output of the OR gate 412 is fed to the shift register 420 timed to the control signal CSC and, in the meantime, the data in the shift registers 402, 404 and 406 are shifted timed to the control signal CSB.

By the operation described so far, the three bits of font data in the first row of the block B8 are fully reduced in the X direction and the result is stored in the shift register 420.

After further shift which is caused by the control signal CSB, the above-described operation is repeated on the font data associated with the block B7 and the reduction data is stored in the shift register 420. Repeating such a procedure eight consecutive times causes Y reduction data associated with the first rows of font data in the blocks B1 to B8 (see FIG. 9) to be stored in the shift register 420. At this time, i.e., a time T2 in FIG. 13, the control signal CSG is produced and the output of the shift register 420 is applied to the register via the OR gates 422A to 422H.

When the same operation is performed on the second row of font data in the blocks B1 to B8, the X reduction data associated with the first row of font data and those associated with the second row are applied to the OR gates 422A to 422H. As a result, the outputs of the OR gates 422A to 422H represent the result of AND operation between the first and second rows of data shown in the middle of FIG. 10.

Thereafter, X reduction data associated with the third row of font data in the blocks B1 to B8 are delivered from the shift register 420. Then, they are subjected to an AND operation together with the above-mentioned operation result so that the OR gates 422A to 422H produce the Y reduction data as shown in the bottom of FIG. 10. These data respectively are routed to the output terminals S1 to S8 via the register 424. Then, the register 424 is cleared by the control signal CSF to allow the foregoing consecutive steps to be repeated on the other rows of the blocks. That is, the foregoing steps are repeated eight times to transform the 24 by 24 dot matrix shown in FIG. 1 to the reduced 8 by 8 dot matrix.

While the illustrative embodiment is constructed to reduce a 24 by 24 dots font to an 8 by 8 dots font, such is only illustrative and any other desired reduction ratio may be set up. In addition, the reduction ratios in the X and Y directions which have been shown and described as being equal are not restrictive and may differ from each other.

Furthermore, the flowchart shown in FIG. 6 and the block diagrams shown in FIG. 12 are merely for illustration and any other suitable procedure and/or circuit arrangement is usable insofar as it is capable of implementing the principles of reduction shown in FIGS. 3A-5B.

In summary, it will be seen that the present invention provides a character font reduction method which eliminates the need for storage of reduced font data in a font memory to thereby save the capacity of the memory and allow a reduced font to be readily attained by means of a simple construction. This advantage is derived from the unique construction wherein font data are reduced based on font data which are stored in a font memory or a character generator in advance.

Even the case wherein reduced fonts are stored in a font memory may advantageously be implemented by the present invention. If a reduced font is generated using a font reduction program and based on an existing font, for example, time and labor which would otherwise be consumed for frequent generation of a reduced font will be saved.

Various modifications will become possible for those skilled in the art after receiving the teachings of the present disclosure without departing from the scope thereof.

What is claimed is:

1. A method of reducing a character font in which font data representative of the character font consists of a dot matrix having a plurality of rows of arranged in a first direction and a plurality of columns of arranged in a second direction perpendicular to the first direction, said method comprising the steps of:

(a) producing first data by dividing one of the rows in the first direction into a plurality of blocks such that each of the blocks includes a predetermined number of dots and, then, converting the one row to unit dots associated with the respective blocks;

(b) producing second data by shifting the row in the step (a) in the first direction to constitute a shifted row, then dividing the shifted row into a plurality of blocks, and then converting the shifted row to unit dots associated with the respective blocks;

(c) computing third data by associating the first and second data with each other in the second direction;

(d) computing third, row-by-row data by sequentially performing the steps (a), (b) and (c) in the second direction on the rows of a number which corresponds to the predetermined number;

(e) computing fourth data by associating in the second direction the third, row-by-row data computed by the step (d);

(f) computing the fourth data associated with the whole character font by repeating the consecutive steps (a) through (e); and

(g) converting the respective blocks consisting of dots associated with the plurality of fourth data computed by the step (f) to unit dots associated with the respective blocks.

2. A method as claimed in claim 1, wherein step (a) comprises a step (h) of, when any of the rows of each of the blocks includes more than a predetermined number of dots immediately next to each other, making the unit dot of said block a black dot and, when otherwise, deciding that printing is needless.

3. A method as claimed in claim 1, wherein step (b) comprises a step (h) of, only when any of the rows in the first direction extends over two nearby blocks, shifting said row one dot in the first direction.

4. A method as claimed in claim 1, wherein the steps (c) and (d) comprise a step (h) of computing each of the third data by providing an AND of the first data and the second data.

5. A method as claimed in claim 4, wherein the steps (e) and (f) comprise a step (i) of computing the each of the fourth data by providing an AND of the respective third data in the second direction, and a step (j) of deciding an AND which is a ONE as a black dot.

* * * * *